International Journal of Industrial Engineering & Production Management (2015)
September 2015, Volume 26, Number 3
pp. 329-344

http://IJIEPM.iust.ac.ir/

MODELING AND A SOLUTION APPROACH FOR THE UNIVERSITY COURSE TIMETABLING PROBLEM AND FACULTY-COURSE ASSIGNMENT: A CASE STUDY

Farin Rastgar-Amini & Seyed Hamid Mirmohammadi*

Farin. Rastgar-Amini, *Msc Student, Department of Industrial and Systems Engineering, Isfahan University of Technology*
Seyed. Hamid. Mirmohammadi, *Assistant Professor, Department of Industrial and Systems Engineering, Isfahan University of Technology*

Keywords

Timetabling,
Linear Binary-Programming,
Case Study,
Conflict Matrix,
Ant Colony System,
Simulated Annealing

ABSTRACT

Having the number of students pre-enrolled jointly in each combination of two courses and considering the faculty time and course preferences, we investigate the problem of minimizing the number of synchronous courses with common students. This minimization is performed in proportion to the amount of their common students using a multi objective LP model with 0-1 variables. In this way, all courses, including 2&3-units courses are scheduled in a time-slot of 3-units courses. Scheduling all courses in the time-slots of 3-units courses decreases the number of variables and constraints in the MIP model considerably but, it omits some alternative solutions in which the 2-units courses are scheduled differently. To consider these alternative solutions, another LP model is presented which takes the solution obtained by the first model as a data entry and improve it via moving the 2-units courses during other time-slots. Since course timetabling problems belong NP-Hard problems, the efficiency of the proposed method decreases trough increasing the size of problem. Therefore, two metaheuristic algorithms, Ant Colony System (ACS) algorithm and Simulated Annealing (SA) algorithm are presented. In order to demonstrate the performance of the proposed algorithms, we compared the efficiency of algorithms both in small and large instances of the problem via solving the test problems provided in industrial engineering department of IUT. In small scale, the ACS algorithm has an average error of 1.08 % from the optimality and this number is 1.82% for SA. In large scale, the computational time of the ACS algorithm increases considerably more than SA algorithm while the average value of the objective function of SA has no tangible difference than ACS.

نشــــریه بین المللـــی مهنـــدسی صنـــایع و مدیــــریت تولیـــد

شمـــاره ۳، جلـــد ۲۶، پاییز ۱۳۹۴

صفحــــه ۳۴۴-۳۳۰

نشریه بین المللی مهندسی صنایع و مدیریت تولید
دانشگاه علم و صنعت ایران

N: 2008-4870

http://IJIEPM.iust.ac.ir/

زمانبندی دروس دانشگاهی و تخصیص استاد – درس - زمان براساس آمار ثبت‌نام مقدماتی، مطالعه موردی

فرین راستگار امینی و سید حمید میرمحمدی*

کلمات کلیدی

برنامه‌ریزی آموزشی،
مدل‌سازی اعداد صحیح صفر و یک،
ماتریس تلاقی،
سیستم اجتماع مورچگان،
شبیه‌سازی تبرید

چکیده:

با در دست داشتن آمار متقاضیان هر درس از طریق ثبت‌نام مقدماتی دروس، یک مدل ریاضی صفر و یک برای مسئله زمانبندی دروس دانشگاهی در این مقاله ارائه می‌شود. در این مدل ترجیحات اساتید در مورد بازه‌های زمانی و موضوعات درسی و همچنین دسترس‌پذیری دروس برای دانشجویان جهت اخذ، در نظر گرفته می‌شود. ابتدا کلیه دروس اعم از دو یا سه واحدی در بازه‌های زمانی متناسب با دروس سه واحدی برنامه‌ریزی می شوند. این امر تعداد متغیرها و محدودیت‌های مسئله را به شدت کاهش می‌دهد. سپس جواب بدست آمده، به عنوان داده‌ی ورودی مدل‌های خطی صفر و یک جدید قرار می‌گیرد که در این مدل‌ها با جابایی مجدد دروس دو واحدی در بازه های مجاز، جواب حاصل ارتقاء یابد. با توجه به اینکه مسئله زمانبندی دروس متعلق به مسائل رده پیچیدگی NP-complete می‌باشند، دو الگوریتم فراابتکاری سیستم اجتماع مورچگان و شبیه‌سازی تبرید برای حل این مسئله ارائه گردیده است. کارایی نسبی الگوریتم‌های پیشنهادی با استفاده از داده‌های دانشکده مهندسی صنایع دانشگاه صنعتی اصفهان و داده-های تصادفی در ابعاد کوچک با مقایسه‌ی جواب‌های بهینه و در ابعاد بزرگ با مقایسه‌ی دو الگوریتم پیشنهادی نشان داده شده است. در ابعاد کوچک میانگین خطای الگوریتم مورچگان نسبت به جواب بهینه برابر ۱/۰۸٪ و میانگین خطای الگوریتم شبیه‌سازی تبرید نسبت به جواب بهینه برابر ۱/۸۲٪ محاسبه شده است. در مورد مسائل با ابعاد بزرگ با افزایش ابعاد مسئله زمان حل الگوریتم مورچگان نسبت به الگوریتم شبیه‌سازی تبرید به طور چشمگیری افزایش می‌یابد در حالیکه میانگین تابع هدف الگوریتم شبیه‌سازی تبرید انحراف قابل ملاحظه‌ای نسبت به الگوریتم مورچگان ندارد.

۱. مقدمه

زمانبندی آموزشی[1] یک فرایند تصمیم‌گیری است که همواره در مراکز آموزشی مطرح می‌باشد. وظیفه‌ی این فرایند، تخصیص منابع محدود به فعالیت‌های موجود در بازه‌های زمانی معین با اهداف خاصی است.

تاریخ وصول: ۹۱/۱۲/۱۲
تاریخ تصویب: ۹۲/۰۹/۰۶

فرین راستگار امینی، دانشجوی کارشناسی ارشد، دانشکده مهندسی صنایع و سیستم‌ها، دانشگاه صنعتی اصفهان، *f.rastgaramini@in.iut.ac.ir*

*نویسنده مسئول مقاله: **دکتر سید حمید میرمحمدی**، استادیار، دانشکده مهندسی صنایع و سیستم‌ها، دانشگاه صنعتی اصفهان، *h_mirmohammadi@cc.iut.ac.ir*

منابع شامل اساتید، اتاق‌ها، آزمایشگاه‌ها و تجهیزات آموزشی مرکز آموزشی است. موضوعات درسی مختلف که طبق برنامه‌ریزی هر مرکز آموزشی باید به دانشجویان ارائه شود، از جمله فعالیت‌های موجود می‌باشد. بهره‌برداری کارا از منابع و برنامه‌ریزی مناسب دروس به منظور افزایش رضایت سازمان آموزشی، اساتید و دانشجویان، اهداف اساسی در زمانبندی آموزشی است.

در طول چند دهه‌ی گذشته تحقیقات بسیاری در زمینه‌ی مسئله‌ی زمانبندی دروس دانشگاهی (UCTP)[2] صورت گرفته است. مدل‌های زیادی در قالب مدل‌های ریاضی،کامپیوتری ،گراف ها و غیره در این زمینه گسترش یافته است. هدف تمام این مدل‌سازی-ها ارائه‌ی برنامه‌ای است که با واردکردن اطلاعات جدید، برنامه‌ی بهینه‌ی متناسب با اهداف ، نیازها و منابع سازمان ساخته شود. در

این زمینه مدل‌های ریاضی به خصوص مدل‌های خطی از جایگاه ویژه‌ای برخوردارند. دیموپولو و میلیوتیس [۱] در سال ۲۰۰۱ یک مدل برنامه‌ریزی عدد صحیح برای تخصیص دروس به بازه‌های زمانی و اتاق‌ها ارائه داده‌اند. هدف مدل ارائه شده، بیشینه نمودن مطلوبیت تخصیص برنامه‌های آموزشی به بازه‌های زمانی است. برنامه‌ی آموزشی دانشجویان هم‌رده مشخص است. بنابراین دروس هر برنامه آموزشی باید در بازه‌های زمانی متفاوت برنامه‌ریزی شوند. داسکالاکی و همکاران [۲] در سال ۲۰۰۴ یک فرمولبندی برنامه‌ریزی عدد صحیح صفر و یک برای مسئله UCTP ارائه داده است. در این تحقیق مطلوبیت تخصیص دروس به بازه‌های زمانی مختلف متفاوت است، لذا داسکالاکی برای تخصیص دروس به برخی از بازه‌های زمانی هزینه‌ای قرار داده است و در صدد کمینه نمودن این هزینه می‌باشد. داسکالاکی و بیرباس [۳] در ادامه تحقیقات خود در سال ۲۰۰۵ یک رویکرد حل برای مدل [۲] ارائه داده است. این رویکرد بر مبنای آزادسازی محدودیت‌های با محاسبات زیاد، بنا نهاده شده است. روش پیشنهادی مدل برنامه-ریزی عدد صحیح قبلی خود را در دو مرحله حل می‌کند. در مرحله اول دروس چند جلسه‌ای بدون در نظر گیری شرط متوالی بودن جلسات به بازه‌های زمانی اختصاص داده می‌شوند. در مرحله دوم دروس بگونه‌ای جابجا می‌شوند که شرط توالی برقرار شود. زمان محاسبات این رویکرد در مقایسه با مطالعه قبلی بسیار کمتر است و تابع هدف آن بسیار نزدیک به تابع هدف مدل یک مرحله‌ای بدست آمده است. در نتیجه رویکرد پیشنهادی عملکرد بهتری داشته است. الیعقوب و همکاران [۴] در سال ۲۰۰۶ یک مدل برنامه ریزی ریاضی عدد صحیح برای تخصیص اساتید به دروس با فرض زمانبندی مشخص دروس ارائه دادند. هدف این مدل، مینیمم کردن نارضایتی فردی و جمعی اساتید بطور فردی و جمعی است. در این مقاله نارضایتی بصورت تابعی از تخصیص اساتید به بازه‌های زمانی و دروس اندازه‌گیری شده است. اسماعیل‌اُوا و همکاران [۵] در سال ۲۰۰۷ یک مدل برنامه‌ریزی صفر و یک چند هدفه برای مسئله تخصیص استاد- درس- زمان (FCTA[۳]) ارائه داده‌اند. اولویت‌های اساتید و سازمان آموزشی در این مسئله مورد توجه بوده است. تعداد ساعت‌ها و دروس مطلوب اساتید در این مدل بیشینه می‌گردد. برای وزن‌دهی به اهداف مختلف و متضاد مطرح در این مسئله از روش AHP[۴] و ANP[۵] استفاده شده است. بورک و همکاران [۶] در سال ۲۰۱۲ برای مسئله‌ی زمانبندی دروس بر مبنای برنامه‌ی آموزشی (CBCT[۶]) که در دومین مسابقه بین‌المللی زمانبندی آموزشی در سال ۲۰۰۷ مطرح شده است، یک فرمولبندی برنامه‌ریزی عدد صحیح و یک روش شاخه و برش ارائه داده‌اند. در این روش برش‌هایی به تعداد نمایی جهت تضمین بهینگی جواب حاصل انجام می‌گیرد. لاچ و لوبیک [۷] مسئله‌ی CBCT را از سه بعد (درس- زمان- اتاق) به دو بعد کاهش داده‌اند.

مولفان به جای حل مدلی با متغیرهای صفر و یک که دروس را به بازه‌های زمانی و اتاق‌ها اختصاص دهد، مسئله را به دو مرحله تقسیم کرده است. در مرحله اول دروس فقط به بازه‌های زمانی تخصیص می‌یابند. در مرحله دوم اتاق‌ها با توجه به فضای شدنی، به زوج‌های درس- زمان اختصاص پیدا می‌کنند. هائو و بنلیک [۸] در سال ۲۰۱۲، رویکرد دیگری برای بدست آوردن حدود پایین برای مسئله‌ی [۶] ارائه داده‌اند. در این روش از جستجوی ممنوع تکراری برای تقسیم کردن مسئله به چند زیر مسئله استفاده شده است، بطوریکه زیرمسئله‌های بدست آمده، قابل حل توسط حل کننده‌های مسائل برنامه‌ریزی خطی عدد صحیح باشند.

مسائل زمانبندی دروس دانشگاهی عمدتاً به مسائل رده‌ی پیچیدگی NP-complete تعلق دارند [۹]. تحقیقات گسترده‌ای در زمینه الگوریتم‌های فراابتکاری انجام شده است. کرزیستوف و همکاران[۱۰] دو الگوریتم مورچگان - سیستم اجتماع مورچگان و سیستم MAX-MIN مورچگان- برای مسئله زمانبندی دروس دانشگاهی ارائه داده‌اند. نتایج بدست آمده با سایر روش‌های جستجوی همسایگی مقایسه شده‌اند. نشان داده شده است که الگوریتم مورچگان تاثیر بسزایی در عملکرد الگوریتم داشته است. یک رویکرد دیگر توسط بورک و همکاران [۱۱] در سال ۲۰۰۳ ارائه شده است. در این مقاله، نویسندگان از الگوریتم سیل عظیم دوک[۷] برای بهبود جواب‌های بدست‌آمده از تخصیص اولیه دروس استفاده کرده‌اند. در اصل، سیل عظیم نوعی الگوریتم جستجوی همسایگی است که روش آن مشابه جستجوی ممنوع و شبیه‌سازی تبرید است. کوستاچ [۱۲] در سال ۲۰۰۵ نیز از روش فراابتکاری شبیه‌سازی تبرید به عنوان سازنده‌ی اصلی الگوریتم خود استفاده کرده است. این الگوریتم از روش رنگ‌آمیزی گراف[۸] برای بدست آوردن یک جواب شدنی استفاده می‌کند. سپس با استفاده از شبیه‌سازی تبرید و با مرتب سازی مجدد بازه‌های زمانی و جابجا کردن دروس بین بازه‌های زمانی، جواب‌های حاصل را ارتقا می‌دهد. چیاراندینی و همکاران [۱۳] در سال ۲۰۰۶ یک الگوریتم فراابتکاری دیگر گسترش داده‌اند. ابتدا با استفاده از روش‌های ابتکاری سازنده، جستجوی محلی و جستجوی ممنوع یک جواب اولیه‌ی شدنی تولید می‌شود. سپس با استفاده از جستجوی همسایگی و روش شبیه‌سازی تبرید جواب بدست آمده را بهبود می‌دهد. هاکان آلاداق و همکاران [۱۴] در سال ۲۰۰۹ یک روش فراابتکاری بر اساس جستجوی ممنوع ارائه داده‌اند. در این تحقیق ساختارهای مختلف همسایگی معرفی شده‌اند. مسئله زمانبندی دروس دپارتمان آمار دانشگاه حاجت‌تپه[۹] با چهار ساختار مختلف همسایگی حل شده و نتایج بدست آمده از ساختارهای مختلف مقایسه شده است. لو و هاوو [۱۵] در سال ۲۰۱۰ یک الگوریتم جستجوی ممنوع انطباقی برای حل مسئله‌ی CBCT ارائه داده‌اند. الگوریتم پیشنهادی از یک چارچوب کلی تمرکزدهی و تنوع بخشی

پیروی می‌کند. ابتدا با استفاده از یک روش ابتکاری حریصانه یک جواب اولیه تولید می‌شود. سپس مرحله انطباقی مرکب از تمرکزدهی و تنوع‌بخشی برای بهبود جواب شروع می‌شود. در این مرحله محدودیت‌های مسئله نقض نمی‌شوند و مسئله همچنان شدنی باقی می‌ماند. شیائو [۱۶] در سال ۲۰۱۱ یک الگوریتم بهینه‌سازی گروه ذرات هیبریدی برای حل مسئله‌ی UCTP ارائه کرده است. عبدالله و ترابیه[۱۷] در سال ۲۰۱۲ یک رویکرد ممتیک[10] با ترکیب الگوریتم تکاملی و جستجوی ممنوع گسترش داده‌اند. این الگوریتم، مجموعه‌ای از ساختارهای همسایگی را با هدف دستیابی به بهبودهایی در کیفیت جواب بکار می‌گیرد.

معیار ارزیابی زمانبندی در تحقیقات صورت گرفته بیشتر به در نظرگیری اولویت اساتید (اولویت زمانی و اولویت درسی) و رعایت محدودیت‌هایی مانند تعداد و ظرفیت کلاس‌ها، تجهیزات آموزشی و غیره محدود میشود. با مکانیزه شدن سیستم‌های ثبت نام الکترونیکی دروس، امکان دستیابی به آمار پیش ثبت‌نام و متقاضیان ثبت‌نام هر درس وجود دارد. یکی از اهداف مدل ارائه شده در این مقاله افزایش قابلیت اخذ دروس از طریق کمینه سازی تعداد دروس همزمان با دانشجویان متقاضی مشترک است، چرا که رعایت این ویژگی در برنامه ریزی دروس هم رضایتمندی دانشجویان در اخذ بیشتر دروس را افزایش می دهد و هم دانشگاه را در رسیدن به اهداف میان مدت خود از جمله دانش آموختگی به هنگام دانشجویان، یاری می‌رساند. اولویت زمانی و اولویت موضوعات درسی اساتید به عنوان دومین تابع هدف و به عنوان جزئی جداناپذیر از برنامه ریزی دروس دانشگاه، در نظر گرفته شده است.

در بخش ۲، فرضیات مسئله عنوان می‌شود و در بخش ۳ مدل ریاضی صفر- یک (BIP[11]) دو مرحله‌ای برای مسئله زمانبندی آموزشی با اهداف ذکر شده ارائه می‌شود. در بخش ۴ یک الگوریتم پیشنهادی بر اساس الگوریتم اجتماع مورچگان و یک الگوریتم بر اساس الگوریتم شبیه‌سازی تبرید برای حل مدل ارائه شده، تشریح شده است. نتایج محاسباتی حاصل از حل مدل ریاضی و روش‌های پیشنهادی و مقایسه آنها در بخش ۵ آورده شده است. نهایتاً در بخش ۶ به نتیجه‌گیری و جمع‌بندی مطالب بیان شده پرداخته می‌شود.

۲. فرضیات و تعریف مسئله

مسئله‌ی ارائه شده در این مطالعه (FCTA)، بر مبنای مطالعه‌ی موردی در دانشکده‌ی مهندسی صنایع دانشگاه صنعتی اصفهان و بر اساس اصول و ضوابط جاری در این مرکز آموزشی، طراحی شده است. بر اساس ثبت‌نام مقدماتی از دانشجویان، تعداد دانشجویانی که به طور مشترک، متقاضی ثبت‌نام در هر دو درس می‌باشند در یک ماتریس به نام ماتریس تلاقی جمع آوری می شود. از اینجا به بعد به آن تعداد از دانشجویانی که نمی‌توانند به دلیل همزمانی ساعت کلاسهای دو درس، آن دو درس را در یک ترم اخذ نمایند، تعداد تلاقی آن دو درس گفته می شود. بدیهی است که تعداد تلاقی دو درس در صورت غیر همزمان شدن کلاسهایشان صفر است، خواه تعداد دانشجویان مشترک آن دو صفر باشد خواه نه. اولویت‌ها و ترجیحات اساتید در مورد دروس و زمان کلاس‌ها نیز توسط آنها با تکمیل فرم های مرتبط فراهم می شود. هفت فرض مهم با توجه به قوانین آموزشی سازمان‌های آموزشی در این مطالعه در نظر گرفته شده است:

۱. دروس دو واحدی طی یک جلسه‌ی دو ساعته و دروس سه واحدی از دو جلسه‌ی یک ساعت و نیمه در هفته تشکیل می‌شود.
۲. تمامی جلسات دروس باید در بازه‌های زمانی تعیین شده توسط سازمان آموزشی برنامه‌ریزی شود. از قرار دادن دروس در روزهای چهارشنبه و اضافه‌کاری برای اساتید پرهیز می‌شود. جدول زمانی مورد نظر مطابق جدول (۱) است.

روز / ساعت	۸ - ۹:۳۰	۹:۳۰ – ۱۱	۱۱ - ۱۲:۳۰	۱۲:۳۰	۱۵	۱۶:۳۰
شنبه	t_1	t_2	t_3	ناهار و اقامه	t_4	t_5
یک‌شنبه	t_6	t_7	t_8	ناهار و اقامه	t_9	t_{10}
دوشنبه	t_{11}	t_{12}	t_{13}	ناهار و اقامه نماز	t_{14}	t_{15}
سه‌شنبه	t_{16}	t_{17}	t_{18}	ناهار و اقامه نماز	t_9	t_{20}

جدول ۱. جدول زمانی روزهای کاری

۳. جلسات دروس سه واحدی باید به دلایل ترجیحات اثربخشی آموزشی به صورت متقارن و به فاصله‌ی ۴۸ ساعت از هم برنامه ریزی شوند. به عنوان نمونه اگر یک جلسه از یک درس سه واحدی به بازه‌ی زمانی ۹:۳۰-۸ روز شنبه تخصیص یابد، جلسه‌ی دوم آن درس باید به بازه‌ی زمانی ۹:۳۰-۸ روز دوشنبه تخصیص یابد.
۴. تعداد دروس تخصیص یافته به هر بازه‌ی زمانی نباید از تعداد اتاق‌های موجود در آن بازه‌ی زمانی تجاوز کند.

۵. به هر استاد در هر بازه‌ی زمانی حداکثر یک درس و به هر درس یک استاد را می‌توان تخصیص داد.

۶. دروس باید به اساتیدی تخصیص یابند که در حیطه‌ی تخصص آن اساتید باشد.

۷. اگر استادی در بازه‌های زمانی خاصی در دسترس نباشد، نباید در آن بازه‌های زمانی، درسی به آن استاد تخصیص داده شود.

۳. مدل‌های ریاضی صفر و یک

در این بخش، دو مدل ریاضی صفر و یک طی یک رویکرد دو مرحله‌ای ارائه می‌شود: در مرحله‌ی اول، تمام درس‌های دو واحدی و سه واحدی در دو روز اول هفته برنامه‌ریزی می‌شوند و تخصیص اساتید نیز انجام می‌گیرد. بدین ترتیب جلسه‌ی اول دروس سه واحدی و تمام دروس دو واحدی در روزهای شنبه و یکشنبه برنامه‌ریزی می‌شوند. با توجه به فرض سوم در این مسئله، جلسه‌ی دوم دروس سه واحدی ، عیناً مطابق روزهای شنبه و یک‌شنبه، در بازه‌های زمانی متناظر روزهای دوشنبه و سه‌شنبه نیز برنامه‌ریزی می‌شود. این امر باعث می‌شود که ضمن رعایت فرض سوم تعداد متغیرهای تصمیم و محدودیت‌های مربوط به مدل به نصف کاهش یابد. بنابراین در مرحله‌ی اول جدول زمانی به صورت جدول (۲) است.

روز / ساعت	۸۹:۳۰ - ۹:۳۰	۹:۳۰-۱۱ — ۱۱	— ۱۲:۳۰	— ۱۳:۳۰	— ۱۵	— ۱۶:۳۰
شنبه	t_1	t_2	t_3	ناهار و اقامه	t_4 t_4	t_5 t_5
یک‌شنبه	t_6 t_6	t_7 t_7	t_8 t_8	ناهار و اقامه	t_9 t_9	t_{10} t_{10}

جدول ۲. جدول زمانی دو روز اول هفته

تا اینجا تخصیص دروس سه واحدی بطور کامل انجام شده است. در مورد دروس دو واحدی دو نکته وجود دارد، اول اینکه تمام دروس دو واحدی در روزهای شنبه و یکشنبه برنامه‌ریزی شده‌اند در حالی که این امر محدودیت اضافه بر مدل ایجاد می‌کند و ممکن است دستیابی به جواب‌های بهتر را تحت تأثیر منفی قرار دهد. دوم اینکه دروس دو واحدی در بازه‌های یک ساعت و نیمه برنامه‌ریزی شده‌اند اما طبق فرض نیاز به بازه‌های دو ساعته دارند. برای رفع مورد اول مرحلی دوم مدلسازی در نظر گرفته شده است و برای مورد رفع دوم علاوه بر بازه‌ی زمانی یک و نیم ساعته نیم ساعت از بازەز زمانی بعدی نیز به دروس دو واحدی تخصیص داده شده است. نمادهای استفاده شده در مدل ریاضی مرحله‌ی اول به شرح زیر است:

C : مجموعه‌ی تمام موضوعات درسی ($\|C\| = n$)

C_2: مجموعه‌ی دروس دو واحدی ($\|C_2\| = n_2$ ، $C_2 \subset C$) .

P: مجموعه‌ی اساتید ($\|P\| = m$)

T: مجموعه‌ی بازه‌های زمانی شنبه و یک‌شنبه ($\|T\| = 10$)

d_{ik} : تعداد دانشجویانی که در دو درس iو k مشترکاً ثبت نام مقدماتی کرده‌اند($i, k \in C$).

p_{jt}: اولویت بازه‌ی زمانی t از نظر استاد j می‌باشد که p_{jt} کمتر نشان‌دهنده‌ی مطلوبیت بیشتر است. ($j \in P$ ، $t \in T$)

q_{ij}: اولویت درس i از نظر استاد j می‌باشد که q_{ij} کمتر نشان‌دهنده‌ی مطلوبیت بیشتر است. ($j \in P$ ، $i \in C$)

B_j: مجموعه‌ی دروسی که در حیطه‌ی تخصص استاد j قرار ندارد ($j \in P$).

U_j: مجموعه‌ی بازه‌های زمانی که استاد j در دسترس نیست ($j \in P$).

r_t: تعداد اتاق‌های موجود در بازه‌ی زمانی t ($t \in T$).

a_j: اهمیت وزنی استاد j بر اساس پایه و مرتبه‌ی علمی است که a_j بیشتر نشان‌دهنده‌ی پایه و مرتبه‌ی بالاتر است.($j \in P$).

x_{ijt}: یک متغیر صفر و یک است که اگر درس i در زمان t به استاد j تخصیص یابد برابر یک و در غیر اینصورت برابر صفر است ($t \in T$ ، $j \in P$ ، $i \in C$).

v_{ik}: یک متغیر صفر و یک است که اگر درس i و درس k همزمان ارائه شوند، برابر یک و در غیر اینصورت برابر صفر است($i, k \in C$).

۳-۱. مرحله‌ی اول

تابع هدف این مدل از حداقل‌سازی سه جزء تشکیل شده است. جزء اول عدم رضایت اساتید را از بازه‌های زمانی تخصیص یافته به ایشان با توجه به رتبه علمی آنها در نظر می‌گیرد، جزء دوم عدم رضایت از موضوعات درسی اساتید و جزء سوم مجموع تعداد تلاقی ترکیب دوتایی از دروس را محاسبه می‌نماید. روابط (۱) الی (۳) اجزاء تابع هدف را نشان می‌دهد:

$$f_1 = \sum_{j \in P} a_j \sum_{t \in T} p_{jt} \sum_{i \in C} x_{ijt} \quad (۱)$$

$$f_2 = \sum_{j \in P} a_j \sum_{i \in C} q_{ij} \sum_{t \in T} x_{ijt} \quad (۲)$$

$$f_3 = \sum_{\substack{k \in C \\ k < i}} \sum_{i \in C} d_{ik} v_{ik} \quad (۳)$$

ماهیت معیارهای ارزیابی جواب که در قالب اجزائ فوق در تابع هدف ارائه شده‌اند متفاوت است. به دلیل مقیاس‌های متفاوت واحد اندازه‌گیری هر جزء تابع هدف، جمع کردن آن‌ها بدون استانداردسازی بی‌معنی است. جهت رفع این مشکل می‌توان

پارامترهای بکار رفته در تابع هدف را نرمال کرد. فرض کنید $d^{\max}$، $p^{\max}$ و $q^{\max}$ مطابق روابط (۴) الی (۶) تعریف می‌شوند:

$$d^{\max} = \max_{\substack{i,k\in C\\ i\neq k}}\left(d_{ik}\right) \tag{۴}$$

$$p^{\max} = \max_{\substack{j\in P\\ t\in T}}\left(p_{jt}\right) \tag{۵}$$

$$q^{\max} = \max_{\substack{j\in P\\ i\in C}}\left(q_{ij}\right) \tag{۶}$$

در این صورت، پارامترهای نرمال شده‌ی مورد استفاده در اجزائ تابع هدف به صورت d_{ik}^{norm}، p_{jt}^{norm} و q_{ij}^{norm} می‌باشند که به ترتیب مطابق روابط (۷) الی (۹) قابل محاسبه‌اند. با این جایگزینی پارامترها، هر کدام از اجزاء سه‌گانه‌ی تابع هدف اولاً استاندارد شده (بدون دیمانسیون) و ثانیاً در بازه‌ی $[0,1]$ مقدار خواهند گرفت.

$$d_{ik}^{norm} = d_{ik} \Big/ d^{\max}\times\frac{n(n-1)}{2} \tag{۷}$$

$$p_{jt}^{norm} = p_{jt} \Big/ p^{\max}\times m\times 10 \tag{۸}$$

$$q_{ij}^{norm} = q_{ij} \Big/ q^{\max}\times m\times n \tag{۹}$$

مدل ریاضی مرحله‌ی اول به شکل زیر است:

$$\min Z_1 = w_1\cdot\sum_{j\in P}a_j\sum_{t\in T}p_{jt}^{norm}\sum_{i\in C}x_{ijt} + w_2\cdot\sum_{j\in P}a_j\sum_{i\in C}q_{ij}^{norm}\sum_{t\in T}x_{ijt} + w_3\cdot\sum_{\substack{k\in C\\ k<i}}\sum_{i\in C}d_{ik}^{norm}v_{ik} \tag{۱۰}$$

Subject to

$$\sum_{t\in U_j}\sum_{i\in C}x_{ijt} + \sum_{i\in B_j}\sum_{t\in T}x_{ijt} = 0 \qquad \forall j\in P \tag{۱۱}$$

$$\sum_{i\in C}x_{ijt}\le 1,\ \forall t\in T \qquad \forall j\in P \tag{۱۲}$$

$$\sum_{i\in C_2}x_{ijt} + \sum_{i\in C}x_{ij(t+1)}\le 1 \qquad \forall j\in P,\ \forall t\in T-\{t_3,t_5,t_8,t_{10}\} \tag{۱۳}$$

$$\sum_{j\in P}\sum_{t\in T}x_{ijt} = 1 \qquad \forall i\in C \tag{۱۴}$$

$$\sum_{j\in P}\sum_{i\in C}x_{ijt}\le r_t \qquad \forall t\in T \tag{۱۵}$$

$$\sum_{j\in P}x_{ijt} + \sum_{j\in P}x_{kjt} - 1\le v_{ik} \quad \forall t\in T,\ \forall\left(\underset{i\neq k}{i,k}\right)\in C \tag{۱۶}$$

$$\sum_{j\in P}x_{ijt} + \sum_{j\in P}x_{kj(t+1)} - 1\le v_{ik} \quad \forall k\in C,\forall i\in C_2, k\neq i, \forall t\in T-\{t_3,t_5,t_8,t_{10}\} \tag{۱۷}$$

$$x_{ijt}\in\{0,1\} \quad \forall i\in C\ ,\ j\in P\ ,\ \forall t\in T \tag{۱۸}$$

$$v_{ik}\in\{0,1\} \qquad \forall\left(\underset{i\neq k}{i,k}\right)\in C \tag{۲۰}$$

در رابطه‌ی (۱۰)، Z_1 مقدار مجموع موزون نرمال‌شده‌ی توابع هدف است. w_1، w_2 و w_3 وزن هر یک از توابع هدف است که توسط سازمان آموزشی تعیین می‌شود. محدودیت‌های (۱۱) تضمین می‌کند که هر استاد به بازه‌های زمانی و دروس ناممکن خود تخصیص داده نشود. هر استادی در هر بازه‌ی زمانی حداکثر در یک جلسه می‌تواند حضور داشته باشد. این شرط در دسته محدودیت (۱۲) گنجانده شده است. اگر به استادی در زمان t ($t\in T-\{t_3,t_5,t_8,t_{10}\}$)، یک درس دو واحدی تخصیص یابد، نباید در بازه‌ی زمانی $t+1$ درسی به وی تخصیص یابد. چرا که نیم ساعت اضافه دروس دو واحدی در بازه‌ی زمانی بعدی ادامه می‌یابد. دلیل استثنا کردن $\{t_3,t_5,t_8,t_{10}\}$ آنست که این بازه‌ها، بازه‌های انتهایی نیم روز و یا روز می‌باشند و اگر درس دو واحدی در این بازه‌ها قرار بگیرد نیم ساعت اضافه‌ی آن بدون هیچ اشتراکی با بازه‌ی بعدی قابل برگزاری می‌باشد. این فرض در دسته محدودیت‌های (۱۳) در نظر گرفته شده است. دسته محدودیت‌های (۱۱) تضمین می‌کند که هر درس حتماً به یک بازه‌ی زمانی و یک استاد اختصاص داده شود. محدودیت‌های (۱۵) محدودیت تعداد اتاق را در نظر می‌گیرد. طبق این محدودیت، به هر بازه‌ی زمانی حداکثر به تعداد اتاق‌های موجود در آن بازه‌ی زمانی، درس اختصاص می‌یابد. مطابق رابطه‌ی (۱۶) در صورتیکه دو درس i و k در یک بازه‌ی زمانی مشترک قرار بگیرند، متغیر صفر و یک v_{ik} دارای مقدار یک خواهد بود. دروس دو واحدی بازه‌ی زمانی t، ($t\in T-\{t_3,t_5,t_8,t_{10}\}$)، با دروس بازه‌ی زمانی $t+1$ دارای هم‌پوشانی هستند. از این‌رو محدودیت‌های (۱۷) برای تشخیص این همپوشانی از طریق مثبت کردن v_{ik} متناظر، ارائه شده‌اند. محدودیت‌های (۱۸) و (۱۹)، صفر و یک بودن متغیرهای x_{ijt} و v_{ik} را تضمین می‌کند.

۳-۲. مرحله‌ی دوم

در این بخش یک مدل ریاضی صفر و یک برای تکمیل و بهبود جواب بدست آمده از مرحله‌ی اول ارائه می‌شود. همانطور که در بخش ۳-۱ اشاره شد، ارائه‌ی دروس دو واحدی به روزهای شنبه و یک‌شنبه محدود می‌شود که این امر ممکن است منجر به از دست دادن جواب‌های بهتر گردد. لذا لازم است این دروس تا حد ممکن بین تمام روزهای هفته توزیع گردند تا بهترین مقدار تابع هدف حاصل شود. لذا لازم است تصمیم‌گیری شود که کدامیک از دروس دو واحدی باید جابجا شود و به کدام بازه‌ی زمانی منتقل شوند تا بهترین جواب ممکن حاصل شود.

اگر به یک بازه‌ی زمانی، دو درس یک‌جلسه‌ای اختصاص یابد اشکالی در برنامه‌ریزی ایجاد نمی‌کند و می‌توان یکی از دو درس را بدون هیچگونه تغییری در تابع هدف (ویژگی ۱را ببینید) به بازه‌ی متناظرش در دو روز دوم هفته منتقل کرد. ولی اگر سه یا بیشتر از سه درس دو واحدی به یک بازه‌ی زمانی خاص اختصاص یابد، باید این درس‌ها را بین آن بازه و بازه های دیگر در دو روز دوم توزیع کرد به گونه ای که مقدار تابع هدف کمینه شود. از **ویژگی ۱** نتیجه می شود که بهترین بازه زمانی برای انتقال یک درس دو واحدی به دو روز دوم، بازه متناظر با بازه‌ی فعلی در دو روز دوم هفته می‌باشد.

ویژگی ۱ : اگر در جواب بهینه حاصل از مدل ۱، یک درس دو واحدی به بازه متناظرش در دو روز دوم هفته منتقل شود، جواب حاصل از نظر تابع هدف بدتر نمی‌شود.

اثبات: انتقال یک درس دو واحدی از هر بازه ای در دو روز اول به بازه متناظر با بازه‌ی فعلی در دو روز دوم نه تنها مقدار جزء سوم تابع هدف (تعداد تلاقی) را افزایش نمی‌دهد بلکه ممکن است موجب کاهش آن شود چرا که تعداد دانشجویان مشترک این درس با کلیه دروس سه واحدی به دلیل برقراری تقارن در زمان‌های ارائه دروس سه واحدی ثابت می‌ماند. از طرف دیگر در صورتی که این درس دو واحدی با یک درس دو واحدی دیگر دارای تلاقی باشد، در اثر این انتقال این تلاقی صفر می‌شود. بدیهی است که این انتقال در جزء اول تابع هدف، به دلیل رعایت اصل تقارن در اعلام اولویت‌های زمانی توسط اساتید، اثری ندارد. جزء دوم تابع هدف نیز به دلیل ثابت ماندن دروس تخصیص یافته به اساتید ثابت می‌ماند و به این ترتیب حکم ثابت است.

نمادها و متغیرهای به کار رفته در این مدل به شرح زیر است:

C_t' : مجموعه‌ی دروس دو واحدی که در حل مدل اولیه در بازه‌ی زمانی t قرار گرفته‌اند ($t \in T$).

x_i: یک متغیر صفر و یک است که اگر درس i از بازه‌ی زمانی t به بازه متناظرش در دو روز دوم منتقل شود برابر یک و در غیر اینصورت دارای مقدار صفر است ($i \in C_t'$).

y_{ij}: یک متغیر صفر و یک است که در صورتیکه هر دو درس i و j در یک بازه‌ی زمانی قرار گیرند (هر دو منتقل شوند و یا هردو منتقل نشوند) دارای مقدار یک و در غیر اینصورت صفر است ($i,j \in C_t'$).

مدل صفر و یک ارائه شده در این بخش به ازای هر یک از بازه‌های زمانی t که شامل بیش از دو درس دو واحدی هستند حل می‌شود و نتیجه‌ی آن بیانگر اینست که کدامیک از دروس دو واحدیِ بازه‌ی زمانی مربوطه، باید به بازه‌ی زمانی متناظر در دو روز دوم هفته منتقل شود.

$$\min Z_2^t = \sum_{(i,j)} \sum d_{ij} \times y_{ij} \qquad (۲۰)$$

$$x_i + x_j - 1 \le y_{ij} \qquad \forall i,j \in C_t' \qquad (۲۱)$$

$$1 - x_i - x_j \le y_{ij} \qquad \forall i,j \in C_t' \qquad (۲۲)$$

$$x_i \in \{0,1\} \qquad \forall i,j \in C_t' \qquad (۲۳)$$

$$y_{ij} \in \{0,1\} \qquad \forall i,j \in C_t' \qquad (۲۴)$$

طبق تابع هدف (۲۰) تعداد تلاقی‌های بازه‌ی زمانی t، طی انتقال دروس یک‌جلسه‌ای کمینه می‌شود. با توجه به محدودیت‌های (۲۱) و (۲۲) اگر هر دو درس i و j در یک بازه‌ی زمانی قرار گیرند (هر دو منتقل شوند و یا هیچ‌کدام منتقل نشوند)، متغیر y_{ij} مقدار یک می‌گیرد.

در عبارت (۲۳)، متغیر x_{it} ($\forall i \in C_t'$) از نوع صفر و یک , در عبارت (۲۴) متغیر y_{ij} ($\forall i,j \in C_t'$) نیز از نوع صفر و یک تعریف شده است. پس از حل این مدل به ازای هر بازه‌ی زمانی t، میزان کل تلاقی‌ها کمتر و یا مساوی تعداد تلاقی‌های مدل مرحله‌ی اول خواهد بود.

۴. روش‌های حل پیشنهادی:

با توجه بهNP-complete بودن مسائل UCTP ، استفاده از الگوریتم‌های فراابتکاری، زمانی که اندازه‌ی مسئله بزرگ باشد، ضروری است. بر این اساس، یک الگوریتم فراابتکاری ACS و یک الگوریتم فراابتکاری SA برای حل این مسئله ارائه شده است.

۴-۱. الگوریتم فراابتکاری *ACS*

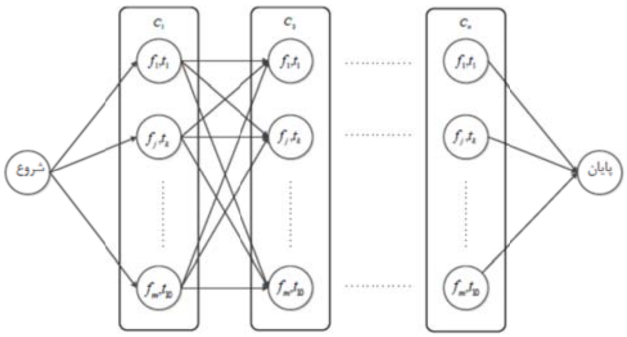
معمولا حل مسائلی که مدل گراف همسایگی آنها براحتی قابل تعریف باشند با الگوریتم مورچگان نتایج مطلوبی بدست می دهد چرا که توسعه این روش بر اساس مدل گرافی مسئله فروشنده دوره گرد بوده است. به عنوان مثال در مسئله مسیریابی وسایل نقلیه مشتریان رئوس گراف و مسیر بین هر دو مشتری یالهای گراف را تشکیل می دهد. به همین دلیل بوده که تقوی فرد و همکاران [۱۹] یک روش اصلاح شده از الگوریتم مورچگان رای برای این مسئله در حالتی که پنجره زمانی برای ملاقات مشتریان وجود دارد ارائه داده اند. از از آنجا که توسعه گراف همسایگی مسئله ارائه شده در این مقاله براحتی امکان پذیر است و در بخش بعدی آورده شده است، الگوریتم مورچگان به عنوان یک کاندید برای روش حل این مسئله در نظر گرفته شده است. در بخش‌های ۴-۱-۱ الی ۴-۱-۶ مفاهیم الگوریتم ACS گسترش داده شده تشریح می‌شود.

۴-۱-۱. گراف همسایگی مسئله

در مرحله‌ی اول باید گراف همسایگی مسئله‌ی تخصیص دروس ($C = \{c_1,...,c_n\}$) به اساتید ($F = \{f_1,...,f_m\}$) و بازه‌های زمانی ($T = \{t_1,...,t_{10}\}$) توسعه داده شود تا بتوان الگوریتم مورچگان متناسب با آن را طراحی نمود. چرا که یکی از ارکان اصلی الگوریتم مورچگان، طراحی گراف نمایش مسئله است بطوریکه خصوصیتی از این گراف (مسیر، پوشش، خوشه، ...)، نشان دهنده‌ی یک جواب مسئله باشد. گراف همسایگی مسئله FCTA در قالب یافتن بهترین مسیر

باشد. گراف همسایگی مسئله FCTA در قالب یافتن بهترین مسیر گسترش داده می‌شود. فرض کنید F_{c_i} مجموعه‌ی اساتیدی است که دارای تخصص تدریس درس c_i می‌باشند. T_{f_j} را نیز در نظر بگیرید مجموعه‌ی بازه‌های زمانی است که استاد f_j در دسترس است. بنابراین مجموعه‌ی $C_i = \bigcup_{f_j \in F_{c_i}} f_j \times T_{f_j}$ شامل تمام زوج مرتب‌های ممکن استاد- زمان (f_j, t_k)، جهت برنامه‌ریزی درس c_i است.

گراف همسایگی مسئله‌ی مورد نظر، یک گراف جهت‌دار چندبخشی[۱۲] متشکل از n بخش بصورت $G = C_1 \cup C_2 \cup \ldots \cup C_n$ است. بخش C_iاُم گراف مربوط به درس c_i است و شامل $\left\| \bigcup_{f_j \in F_{c_i}} f_j \times T_{f_j} \right\|$ گره می‌باشد، بطوریکه هر گره نماینده‌ی زوج‌های استاد- زمان ممکن برای درس c_i است. از هر گره هر بخش از گراف یک یال به سمت هر گره بخش بعدی وجود دارد. اتخاذ چنین سیاستی در ترسیم گراف همسایگی مسئله‌ی FCTA منجر می‌شود که همواره اساتید و بازه‌های زمانی شدنی به دروس تخصیص یابد. شکل (۱) گراف همسایگی مسئله‌ی FCTA را نشان می‌دهد.



شکل ۱. گراف همسایگی مسئله

در هر تکرار از الگوریتم، هر مورچه از گره صفر شروع به حرکت می‌کند و روی بخش‌های گراف حرکت کرده و پس از رسیدن به گره پایان، یک جواب کامل برای تخصیص استاد و زمان به دروس می‌سازد. هر مورچه در هر بخش از گراف یکی از گره‌های بخش بعدی را بر اساس اطلاعات فرومون حاصل از تکرارهای قبلی و به صورت احتمالی انتخاب می‌کند. بدین ترتیب هر درس به یک استاد و یک بازه‌ی زمانی تخصیص می‌یابد. مسئله‌ی طرح شده را می‌توان به مسئله‌ی فروشنده‌ی دوره‌گرد تعمیم یافته (GTSP)[۱۳] کاهش داد.

در هر بازه‌ی زمانی به هر استاد حداکثر یک درس می‌توان تخصیص داد. لذا برای اینکه مورچه‌ها همواره در فضای شدنی جستجو کنند، به محض اینکه زوج مرتب (f_j, t_k) برای درس c_i انتخاب شد، این گره از بین گره‌های بخش‌های $c_{i+1},\ldots,c_n$ حذف می‌شود. به عبارت دیگر بدین ترتیب مجموعه‌ی گره‌های مجاز بعدی مورچه‌ها کاهش می‌یابد. اگر درس c_i عضو مجموعه‌ی دروس دو واحدی باشد دو بازه‌ی زمانی متوالی را اشغال می‌کند. بنابراین علاوه برگره‌های (f_j, t_k)، باید گره‌های (f_j, t_{k+1}) را نیز از بین گره‌های بخش‌های $c_{i+1},\ldots,c_n$ حذف کرد. نحوه‌ی کدینگ جواب‌های مسئله در این الگوریتم یک ماتریس با ابعاد $10 \times (m+1)$ است، بطوریکه سطرهای f_1 تا f_m آن نماینده‌ی اساتید و ستون‌های آن نماینده‌ی بازه‌های زمانی است. درایه‌های متناظر با سطرهای f_1 تا f_m و ستون‌های ماتریس نیز نشان‌دهنده‌ی دروس می‌باشد. سطر آخر ماتریس مقدار تابع هدف جزئی مسئله متناظر با هر یک از بازه‌های زمانی (z_i) را نشان می‌دهد. مجموع توابع هدف جزئی با مقدار کل تابع هدف برابر است. جدول (۳) نمای کلی جواب الگوریتم را نشان می‌دهد. اگر درس به استاد و بازه‌ی زمانی تخصیص یابد، در سلول حاصل از تلاقی ستون و سطر ، درس قرار می‌گیرد.

درس / استاد	t_1	t_2	$\cdots$	t_{10}
f_1	c_1	c_2		
f_2				
$\vdots$			c_i	
f_m				c_n
z	z_1	z_2	$\cdots$	z_{10}

جدول ۳. کدینگ جواب الگوریتم ACS

۴-۱-۲. تولید جواب اولیه

مقدار اولیه فرومون بر اساس مقدار تابع هدف یک جواب اولیه تعریف می‌شود. برای تولید جواب اولیه به ترتیب برای هر درس یکی از زوج‌های استاد- زمان موجود در مجموعه‌ی زوج مرتب‌های امکان‌پذیر برای آن درس (C_i)، انتخاب می‌شود. هر زوج استاد- زمان که برای هر درسی انتخاب شود، از میان زوج‌های استاد-

زمان دروس دیگر حذف می‌گردد. پس از اینکه تخصیص استاد و زمان برای تمام دروس انجام گرفت یک جواب موجه حاصل می‌شود. τ_0 مقدار فرومون اولیه‌ی یال‌ها می‌باشد که مطابق رابطه‌ی (۲۵) محاسبه می‌شود.

$$\tau_0 = \frac{1}{z_{first}} \qquad (۲۵)$$

z_{first} مقدار تابع هدف جواب اولیه است.

۴-۱-۳. قانون احتمال انتخاب

مورچه‌ای که در درس c_{i-1} قرار دارد، زوج استاد- زمان $(f_{j'},t_{k'})$ درس c_i را با توجه به رابطه‌ی (۲۶) انتخاب می‌کند. همانطور که در این رابطه نشان داده شده است، هر مورچه زوج استاد- زمان درس c_i را بر اساس مقادیر فرومون $\tau_{(i-1,j,k),(i,j',k')}$ انتخاب می‌کند.

$$(f_{j'},t_{k'}) = \begin{cases} \arg\max\limits_{\substack{(j,k)\in C_{i-1} \\ (j',k')\in C_i}} (\tau_{(i-1,j,k),(i,j',k')}) & if \quad q \le q_0 \\ (f_j,t_k) & \text{o.w} \end{cases} \qquad (۲۶)$$

در رابطه‌ی فوق، q یک مقدار تصادفی متعلق به بازه‌ی $[0,1]$ و q_0 پارامتر مسئله است. (f_j,t_k)مشخصات تصادفی یک گره از درس c_i است که احتمال انتخاب آن با استفاده از رابطه‌ی (۲۷) بدست می‌آید. ابتدا یک مقدار تصادفی q تولید می‌شود. اگر مقدار تولید شده برای q کمتر و یا مساوی q_0 بود، از مجموعه‌ی یال‌های متناظر با درس c_{i-1} و درس c_i، یالی انتخاب می‌شود که دارای بیشترین مقدار $\tau_{(i-1,j,k),(i,j',k')}$ باشد. در غیر اینصورت احتمال انتخاب هر گره طبق رابطه‌ی (۲۷) محاسبه شده و با استفاده از روش چرخ رولت[۱۴] گره (f_j,t_k) انتخاب می‌شود.

$$P(f_j,t_k) = \frac{\tau_{(i-1,j,k),(i,j',k')}}{\sum\limits_{(j',k')}\sum\limits_{(j,k)} \tau_{(i-1,j,k),(i,j',k')}} \qquad (۲۷)$$

۴-۱-۴. جستجوی محلی

در هر تکرار پس از اینکه تمام مورچه‌ها یک تخصیص کامل از اساتید و بازه‌های زمانی ایجاد کردند جواب تمام مورچه‌ها با استفاده از یک الگوریتم جستجوی محلی بهبود داده می‌شود قبل از اجرای جستجوی محلی لازم است لیست دروس و اساتید تخصیص یافته به هر بازه‌ی زمانی مطابق با ماتریس شکل (۴) مشخص شود. سپس تمام بازه‌ی زمانی امکان‌پذیر دیگر برای هر درس بررسی می‌شود و هر یک از دروس از هر بازه‌ی زمانی که در آن قرار دارد همراه با استاد تخصیص یافته به ترتیب بین تمام بازه‌های زمانی امکان‌پذیر جابجا می‌شودتا بهترین مقدار تابع هدف مسئله حاصل شود.

مثال (۱): جدول (۴) را در نظر بگیرید. در جدول (الف) چهار درس به سه استاد و دو بازه‌ی زمانی تخصیص یافته است و با توجه به ماتریس تلاقی (جدول ب) مقدار کل تابع هدف برابر ۷ است. فرض می‌کنیم استاد f_1 و f_3 در هر دو بازه‌ی زمانی و استاد f_2 تنها در بازه‌ی t_1 در دسترس است.

زمان / استاد	t_1	t_2
f_1	c_1	c_4
f_2	c_2	
f_3		c_3
Z	1	6

(الف)

درس / درس	C_1	C_2	C_3	C_4
C_1	0	1	12	40
C_2	1	0	0	0
C_3	12	5	0	6
C_4	40	0	6	0

(ب)

جدول ۴. (الف) نمایش جواب مثال ۱، (ب) ماتریس تلاقی مثال ۱

ابتدا درس c_1 بررسی می‌شود. چون استاد f_1 در بازه‌ی t_2 درس دیگری ارائه داده است نمی‌توان درس c_1 را جابجا کرد. اگر درس c_2 به بازه‌ی t_2 انتقال یابد مقدار تابع هدف برابر ۶ می‌شود لذا c_2به t_2 منتقل می‌شود. درس c_3 را نمی‌توان جابجا کرد زیرا استاد f_2 تنها در بازه‌ی t_1 در دسترس است.

درس c_4 نیز همانند c_1 امکان جابجایی ندارد. این الگوریتم تا جایی تکرار می‌شود که دیگر بهبودی در تابع هدف حاصل نشود. بدین ترتیب جواب بدست آمده توسط هر مورچه تا جای ممکن بهبود داده می‌شود.

۴-۱-۵. بروز کردن اثر فرومون

بروز رسانی فرومون در دو مرحله انجام می‌گیرد: (۱) بروز رسانی محلی و (۲) بروز رسانی عمومی.

بروز رسانی محلی پس از حرکت هر مورچه از روی هر یال انجام می‌گیرد. قاعده‌ی بروز رسانی محلی باعث می‌شود که مطلوبیت یال‌ها به صورتی پویا در حال تغییر باشد. هر زمان که یالی توسط مورچه‌ای انتخاب شد، مطلوبیت آن توسط قاعده‌ی بروز کردن محلی اندکی کاهش می‌یابد زیرا مقداری از فرومون خود را از دست می‌دهد. این امر موجب می‌شود که از هم‌گرا شدن مورچه‌ها به جستجو در اطراف یک جواب بهینه‌ی محلی جلوگیری شود و مطلوبیت جستجوی یال‌های ملاقات نشده بیشتر شود و در نتیجه فضای جواب بیشتری از مسئله مورد جستجو قرار گیرد.

مقدار فرومون یال‌هایی که توسط مورچگان طی شده است طبق رابطه‌ی (۲۸) بروز می‌شود. $\xi \in [0,1]$ پارامتر کاهش فرومون است.

$$\tau_{(i,j,k),(i+1,j',k')} \leftarrow (1-\xi)\cdot\tau_{(i,j,k),(i+1,j',k')} + \xi\cdot\tau_0 \qquad (۲۸)$$

بروز کردن عمومی زمانی انجام می‌گیرد که تمام مورچه‌ها جواب کاملی ایجاد کرده باشند. در این زمان مقدار فرومون یال‌ها توسط معادله‌ی (۲۹) بروز می‌شود.

$$\tau_{(i,j,k),(i+1,j',k')} \leftarrow (1-\rho)\cdot\tau_{(i,j,k),(i+1,j',k')} + \rho\cdot\Delta\tau \qquad (۲۹)$$

$\rho \in [0,1]$ پارامتر تبخیر فرومون است. در الگوریتم ACS تنها مورچه‌ای که بهترین جواب را ایجاد کرده است، مجاز است فرومون یال‌های متعلق به جواب خود را افزایش دهد. بنابراین مقدار $\Delta\tau$ مطابق رابطه‌ی (۳۰) تعیین می‌شود.

$$\Delta\tau = \begin{cases} \dfrac{1}{Z_{ACS}} & if \;\; (i,j,k),(i+1,j',k') \in best\_in\_iteration \\ 0 & ow \end{cases} \qquad (۳۰)$$

بنابر رابطه‌ی (۳۰) اگر یال متناظر با گره (f_j, t_k) مربوط به درس c_i و گره $(f_{j'}, t_{k'})$ مربوط به درس c_{i+1}، عضوی از مجموعه‌ی یال‌های بهترین جواب بدست آمده در هر تکرار (*best_in_iteration*) باشد، مقدار فرومون آن افزایش می‌یابد.

۴-۱-۶. شرط توقف

شرط توقف الگوریتم ACS را می‌توان حداکثر تعداد تکرار، حداکثر زمان اجرا و حداکثر تعداد تکرارهایی که در تابع هدف بهبودی حاصل نمی‌شود در نظر گرفت. پس از بررسی هر سه روش، شرط توقف به این ترتیب تعریف شده است: اگر در N تکرار آخر بهبودی در جواب تابع هدف حاصل نشود، الگوریتم متوقف می‌شود. مقدار N به ابعاد مسئله‌ی مورد نظر بستگی دارد.

مقادیر پارامترهای الگوریتم ACS مطابق جدول (۵) تنظیم شده است.

جدول ۵. مقادیر پارامترهای الگوریتم ACS

N	q_0	ρ	ξ	تعداد مورچه
۱۵	۰/۹	۰/۱	۰/۱	۱۵

۴-۲. الگوریتم فرا ابتکاری *SA*

الگوریتم شبیه‌سازی تبرید در حل بسیاری از مسائل پیچیده، بسیار موفق و خوب عمل کرده است. به عنوان مثال الگوریتم SA برای حل مسئله‌ی TSP با ۴۰۰ شهر استفاده شده است [۱۸]به همین دلیل بوده است که این روش در دامنه گسترده و متنوعی از مسائل بهینه سازی استفاده شده است. مدیریت مالی و پیش بینی شاخص بورس [۲۰]، زمانبندی پروژه با منابع محدود [۲۱] و کنترل موجودی [۲۲] از جمله این دسته مسائل بهینه سازی می باشند. که همچنان جولانگاه روش شبیه سازی تبرید می باشند. با توجه به اینکه مسئله‌ی FCTA طرح شده در این تحقیق را می‌توان مطابق مطالب بیان شده در بخش ۴-۱ به یک مسئله‌ی یافتن بهترین مسیر کاهش داد، به نظر می‌رسد بکارگیری الگوریتم SA برای حل مسئله مفید باشد. از اینرو یک الگوریتم SA، برای حل مسئله‌ی FCTA گسترش داده می‌شود. مراحل طراحی الگوریتم SA در بخش‌های۰ الی ۴-۲-۵ شرح داده می‌شود.

۴-۲-۱. نحوه‌ی تولید جواب اولیه و تولید همسایگی

برای تولید جواب اولیه، ابتدا همانند روشACS مجموعه‌ی زوج مرتب‌های امکان‌پذیر هر درس تولید می‌شود. . سپس به ترتیب برای هر درس یکی از زوج‌های استاد- زمان موجود در مجموعه‌ی زوج مرتب‌های امکان‌پذیر برای آن درس (C_i)، انتخاب می‌شود. هر زوج استاد- زمان که برای هر درسی انتخاب شود، از میان زوج‌های استاد- زمان دروس دیگر حذف می‌گردد. پس از اینکه تخصیص استاد و زمان برای تمام دروس انجام گرفت یک جواب موجه حاصل می‌شود. نحوه‌ی نمایش جواب همانند الگوریتم ACS است (شکل ۴).

تولید جواب همسایه در این الگوریتم از دو مرحله تشکیل شده است. این دو مرحله به شرح زیر است:

۱. در مرحله‌ی اول، ابتدا دروس و اساتید تخصیص داده شده به هر بازه‌ی زمانی مشخص می‌شود. سپس مطابق شکل (۴) تابع هدف جزئی مربوط به هر یک از بازه‌های زمانی (z_k) محاسبه می‌گردد. بازه‌ی زمانی که دارای بیشترین مقدار تابع هدف

باشد (بازه‌ی زمانی مبدأ) و بازه‌ی زمانی که دارای کمترین مقدار تابع هدف باشد (بازه‌ی زمانی مقصد) انتخاب می‌شود. از بین دروس بازه‌ی زمانی مبدأ، درسی که حذف آن از این بازه‌ی زمانی بیشترین کاهش تابع هدف را بدنبال داشته باشد، برای جابجایی انتخاب می‌شود. در صورتی که استاد مربوطه در بازه-ی زمانی مقصد درس دیگری را ارائه نکرده باشد و درآن بازه‌ی زمانی در دسترس باشد به بازه‌ی زمانی مقصد منتقل می‌شود. اگر انتقال امکان‌پذیر نباشد ، از بازه‌ی زمانی مبدأ، درس بعدی که حذف آن منجر به بیشترین کاهش تابع هدف باشد انتخاب و منتقل می‌شود. اگر انتقال برای هیچ یک از دروس بازه‌ی زمانی مبدأ ممکن نباشد، بازه‌ی زمانی بعدی که دارای بیشترین مقدار تابع هدف جزئی باشد به عنوان بازه‌ی زمانی مبدأ انتخاب می‌شود.

۲. در مرحله‌ی دوم، یک درس بصورت تصادفی انتخاب می‌شود و سپس بازه‌ی زمانی و استاد تخصیص یافته به آن درس با یک بازه‌ی زمانی و استاد دیگر بصورت تصادفی تعویض می‌شود. لازم است که بازه‌ی زمانی و استاد انتخاب شده موجه باشد؛ بدین معنی که استاد مربوطه در آن بازه‌ی زمانی در دسترس بوده و درس دیگری ارائه نکرده باشد. در صورتیکه بازه‌ی زمانی انتخاب شده موجه نباشد، بازه‌ی زمانی دیگری بصورت تصادفی انتخاب می‌شود. چنانچه هیچ بازه‌ی زمانی و استاد موجهی یافته نشود، درس دیگری بصورت تصادفی انتخاب می‌شود.

۴-۲-۲. دمای اولیه و دمای نهایی

مقدار دمای اولیه در یافتن جواب بهینه تأثیر بسزایی دارد. اگر این مقدار بالا انتخاب شود، آنگاه می‌توان گفت که خطر قرارگیری در بهینه‌ی محلی کاهش می‌یابد و اگر این مقدار پایین باشد، خطر قرارگیری در بهینه‌ی محلی وجود دارد. دمای اولیه از طریق انجام یکسری آزمایشات قبل از شروع الگوریتم، محاسبه می‌گردد. به این ترتیب که توسط روش تولید همسایگی،۲۰ جواب جدید تولید می‌شود. میزان تغییر در تابع هدف به ازای دو جواب متوالی (Δf) محاسبه می‌شود. دمای اولیه (T_0) و دمای نهایی (T_f) مطابق رابطه‌ی (۳۱) و (۳۲) محاسبه می‌شود.

$$T_0 = \min\{\Delta f\} + \frac{\max\{\Delta f\} - \min\{\Delta f\}}{10} \quad (۳۱)$$

$$T_f = \min\{\Delta f\} \quad (۳۲)$$

۴-۲-۳. تابع تبرید

در الگوریتم SA پیشنهادی، تابع کاهش دما مطابق رابطه‌ی (۳۳) تعریف می‌شود. T_p در این رابطه بیانگر دما در تکرار pاُم است.

$$T_p = \frac{T_{p-1}}{1+\beta} \quad (۳۳)$$

پارامتر β و متعاقباً پارامتر M مطابق رابطه‌ی (۳۴) و (۳۵) محاسبه می‌شود.

$$\beta = \frac{T_0 - T_f}{M \cdot T_0 \cdot T_f} \quad (۳۴)$$

$$M = 50 \cdot \frac{r(r-1)}{2} \quad (۳۵)$$

r در رابطه‌ی (۳۵) بیانگر ابعاد مسئله است که بر اساس تعداد زوج‌های استاد- زمان ممکن که در بخش ۴-۱-۱ بحث شد تعیین می‌شود و در رابطه‌ی (۳۶) نشان داده شده است.

$$r = \left\| \bigcup_{i=1}^{n} C_i \right\| \quad (۳۶)$$

۴-۲-۴. شرط تعادل

بطو کلی در الگوریتم SA، تعداد جواب‌های پذیرفته شده یا تعداد کل جواب‌های تولید شده در هر دما، به عنوان مبنایی برای بررسی شرط تعادل در آن دما منظور می‌شود. به تعداد تعویض‌ها در هر درجه‌ی حرارت جهت بررسی شرط تعادل، دوره[15] گفته می‌شود. در هر دمایی به اندازه‌ی طول دوره (e)، جواب پذیرفته شده ایجاد می‌شود. سپس نسبت متوسط مقادیر تابع هدف جواب‌های پذیرفته شده در هر دوره با متوسط مقادیر تابع هدف جواب‌های پذیرفته شده در کل دوره‌ها مقایسه می‌شود. اگر این نسبت زیاد باشد، نشان‌دهنده‌ی میزان نوسان در جواب‌های بدست آمده در دوره‌ی جاری است. در نتیجه سیستم به حالت تعادل نرسیده و مجدداً در آن دما، به اندازه طول دوره باید جواب پذیرفته شده ایجاد شود. اگر این مقدار از یک عددی (ε) کوچکتر باشد، به این معناست که شرط تعادل برقرار بوده و دما کاهش پیدا می‌کند و این فرآیند در دمای بعدی تکرار می‌شود. هربار کنترل می‌شود که مجموع تعداد جواب‌های پذیرفته شده در هر دما از حداکثر تعداد جواب‌های پذیرفته شده (M)، بیشتر نباشد. شرط تعادل مطابق رابطه‌ی (۳۷) تعریف می‌شود.

$$\frac{\left|\bar{f_e}(T_p) - \bar{f_g}(T_p)\right|}{\bar{f_g}(T_p)} \le \varepsilon \quad (۳۷)$$

پارامترهای رابطه‌ی (۳۷) به صورت زیر تعریف می‌شوند:

$\bar{f_e}(T_p)$: متوسط مقدار تابع هدف برای جواب‌های پذیرفته شده در هر دوره و در دمای T_p

$\bar{f_g}(T_p)$: متوسط مقدار تابع هدف برای جواب‌های پذیرفته شده در کل دوره‌های قبلی و در دمای T_p

ε: عدد مثبت و کوچک، برای بررسی شرط تعادل در دمای مورد نظر

۴-۲-۵. شرط توقف

در الگوریتم پیشنهادی جهت بررسی شرط توقف از معیار دمای نهایی استفاده می‌شود. اگر دما به دمای نهایی برسد، سیستم به حالت انجماد رسیده و متوقف می‌شود. در غیر اینصورت الگوریتم ادامه پیدا می‌کند.

مقادیر پارامترهای الگوریتم SA مطابق جدول (۶) در نظر گرفته شده است.

جدول ۶. مقادیر انتخاب شده برای پارامترهای الگوریتم SA

e	M	ε
۵	۵۰	۰/۰۰۵

۵. نتایج عددی

در این بخش نتایج حل مسائل نمونه با استفاده از مدل ریاضی صفر و یک و روش‌های ACS و SA ارائه می‌شود. جهت کدنویسی مدل ریاضی صفر و یک از حل کننده‌ی Cplex در نرم افزار GAMS(23.5) استفاده شده است. الگوریتم ACS و SA نیز با استفاده از نسخه 7.8.0.347 نرم‌افزار MATLAB کد شده است. تمام مسائل بر روی یک ابر رایانه با مشخصات CPU 6 Core (2.93 GHz) و RAM (24 GB) اجرا شده است.

۵-۱. ابعاد کوچک

برای حل مسائل با ابعاد کوچک (شامل مسائل واقعی و تصادفی) با استفاده از نرم‌افزار GAMS، محدودیت زمانی ۱۰۰۰۰ ثانیه در نظر گرفته شده است.

الف- مسائل واقعی

داده‌های واقعی به کار گرفته شده در این تحقیق شامل اطلاعات برنامه‌ریزی آموزشی چهار نیمسال تحصیلی دانشکده‌ی مهندسی صنایع و سیستم‌ها در دانشگاه صنعتی اصفهان است. در این مسائل تنها جزء سوم تابع هدف مورد توجه دانشکده بوده است. لذا وزن جزء اول و دوم تابع هدف در حل مسئله برابر صفر در نظر گرفته می‌شود. همچنین به دلیل وجود تنها یک تابع هدف نیازی به نرمال‌سازی توابع هدف وجود ندارد.

جدول (۷) نتیجه‌ی حل مرحله‌ی اول برنامه‌ریزی دروس تمام دروس سه واحدی و دو واحدی در ده بازه‌ی زمانی شنبه و یک‌شنبه برای چهار مسئله‌ی یاد شده را ارائه می‌دهد. جواب بهینه‌ی تمام مسائل در محدودیت زمانی ۱۰۰۰۰ ثانیه با استفاده از حل مدل BIP بدست آمده است. ولی واضح است که زمان حل مسائل زیاد است. حل الگوریتم ACS، موفق به دستیابی به جواب بهینه‌ی تمام چهار مسئله و حل الگوریتم SA منجر به دستیابی جواب بهینه‌ی سه مسئله شده است. میانگین درصد خطای الگوریتم ACS برابر با ۰/۰۰٪ و میانگین درصد خطای الگوریتم SA برابر با 1.08٪ است. به طور میانگین الگوریتم ACS و SA به ترتیب ۷۲۳۶ و ۷۲۴۶/۷۵ ثانیه صرفه‌جویی در زمان حل مسائل نسبت به BIP به همراه داشته‌اند.

پس از برنامه‌ریزی دروس در روزهای شنبه و یک‌شنبه لازم است که مدل ریاضی مرحله‌ی دوم برای توزیع دروس دو واحدی بین تمام روزهای شنبه تا چهارشنبه حل شود. همانطور که در بخش سوم ذکر گردید مدل ریاضی مرحله‌ی دوم برای بهبود و تکمیل جواب بدست آمده از مرحله‌ی اول به کار گرفته می‌شود.. در جواب‌های بدست آمده از مرحله‌ی اول، تنها در مورد مسئله‌ی اول حل مدل مرحله‌ی دوم منجر به کاهش تابع هدف به اندازه‌ی ۳ واحد با زمان حل ۰/۰۱ ثانیه شده است. دروس دو واحدی در جواب مسائل دوم الی چهارم به‌گونه‌ای برنامه‌ریزی شده‌اند که به تعداد کمتر از ۳ درس دو واحدی در هر بازه‌ی زمانی قرار گرفته‌اند و مطابق ویژگی ۱ نیازی به حل مدل ریاضی مرحله‌ی دوم وجود ندارد.

جدول ۷. نتایج حل مرحله‌ی اول مسائل نیمسال دوم ۱۳۹۰-۱۳۸۹ الی نیمسال اول ۱۳۹۲-۱۳۹۱ توسط GAMS، ACS و SA

مشخصات مسئله			نرم‌افزار GAMS		الگوریتم ACS			الگوریتم SA		
شماره مسئله	تعداد درس	تعداد استاد	مقدار تابع هدف	زمان حل (ثانیه)	مقدار تابع هدف	زمان حل (ثانیه)	درصد انحراف تابع هدف	مقدار تابع هدف	زمان حل (ثانیه)	درصد انحراف تابع هدف
۱	۴۱	۱۸	۱۲	۶۸۵۳	۱۲	۵۶	۰/۰۰	۱۲	۴۵	۰/۰۰
۲	۴۰	۲۱	۷	۹۳۶۹	۷	۴۸	۰/۰۰	۷	۳۷	۰/۰۰
۳	۴۱	۲۱	۲۳	۷۴۶۹	۲۳	۵۷	۰/۰۰	۲۴	۵۲	۴/۳۵
۴	۴۰	۲۵	۶	۵۴۹۶	۶	۶۱	۰/۰۰	۶	۶۷	۰/۰۰
میانگین			۱۲	۷۲۹۷	۱۲	۶۱	0.00	۱۲/۵	۰/۰۰	1.08

به طور کلی می‌توان در مورد چهار مسئله‌ی حل شده توسط سه رویکرد ارائه شده در تحقیق اینگونه نتیجه گرفت که جواب بهینه‌ی مسائل با محدودیت زمانی ۱۰۰۰۰ ثانیه بدست آمده است، اما الگوریتم‌های ACS و SA ارائه شده نیز به ترتیب با میانگین خطای ۰/۰۰٪ و ۱/۰۸٪ نسبت به جواب بهینه در مدت زمان بسیار کمتر قادر به حل مسائل بوده‌اند. مقادیر تابع هدف بدست آمده

توسط روش‌های حل پیشنهادی تفاوت چشمگیری با مقادیر تابع هدف حل دستی این مسائل توسط کارشناس آموزش دانشکده دارد. به عنوان مثال تعداد تلاقی جدول زمانبندی دروس مسئله یک برابر با ۳۰۱ است که نسبت به جواب بهینه ۳۲۴۴٪ انحراف دارد.

ب- مسائل تصادفی

سیزده دسته مسئله با ابعاد متفاوت جهت بررسی کارایی روش‌های حل پیشنهادی در حل مسائل با ابعاد کوچک تولید شده است. وزن هر یک از اجزاء تابع هدف به ترتیب برابر ۶ ، ۲ و ۱ در نظر گرفته شده است.

برای تولید هر مسئله ابتدا حداکثر تعداد اساتیدی که ممکن است متقاضی درس c_i باشند (A)، تعیین می‌شود. سپس برای هر درس c_i ، یک عدد تصادفی r_i از فاصله‌ی اعداد طبیعی $[1,A]$ انتخاب می‌شود. سپس r_i استاد بطور تصادفی از میان m استاد برای تدریس c_i انتخاب می‌شود. پس از تکمیل این مرحله برای تمام n درس، مجموعه‌ی دروسی که در حیطه‌ی تخصص هر استاد نباشد (B_j) قابل استخراج است. برای اولویت‌دهی دروس، اعداد مجموعه‌ی جایگشت تصادفی $\{1,2,\ldots,n-\|B_j\|\}$ به عنوان اولویت درس iاُم برای استاد j اُم (p_{ij})اختصاص داده می‌شود.

جهت ایجاد مجموعه‌ی بازه‌های زمانی در دسترس برای استاد j اُم، یک عدد تصادفی (r'_j) از فاصله‌ی اعداد طبیعی $[1,10]$ انتخاب می‌شود. سپس r'_j بازه‌ی زمانی بطور تصادفی از میان ده بازه‌ی زمانی به عنوان زمان‌های در دسترس استاد j اُم انتخاب می‌شود. بدین ترتیب می‌توان زمان‌های غیر قابل دسترس استاد j اُم (U_j) را نیز مشخص کرد. برای اولویت‌دهی بازه‌های زمانی، یک عدد تصادفی از فاصله‌ی اعداد طبیعی $[1,4]$ به عنوان اولویت بازه‌ی زمانی tاُم برای استاد j اُم (q_{jt}) به هر بازه‌ی زمانی اختصاص می‌یابد.

جهت تولید ماتریس تلاقی، یک ماتریس متقارن $n \times n$ ایجاد می‌شود. عناصر قطر اصلی این ماتریس برابر صفر است. دیگر عناصر ماتریس تلاقی بر اساس الگوی داده‌های واقعی در دسترس مشخص می‌شود.

در هر دسته، چهارده مسئله و در مجموع ۱۸۲ مسئله‌ی نمونه حل شده و میانگین توابع هدف و میانگین زمان حل مسائل در جدول (۸) گزارش شده است. واضح است که با افزایش ابعاد مسئله زمان حل مسائل با استفاده از نرم‌افزار GAMS به شدت افزایش می‌یابد.

جدول ۸. نتایج حل مسائل تصادفی با ابعاد کوچک با استفاده از GAMS، ACS و SA

مشخصات مسئله			نرم‌افزار GAMS		الگوریتم ACS			الگوریتم SA		
دسته مسئله	تعداد درس	تعداد استاد	میانگین توابع هدف	میانگین زمان حل	میانگین توابع هدف	میانگین زمان حل	درصد انحراف	میانگین توابع هدف	میانگین زمان حل	درصد انحراف
۱	۲۰	۱۰	۰/۳۰۳	۴/۱۴	۰/۳۰۳	۱۴/۲۳	۰/۰۰	۰/۳۰۳	۲/۰۲	۰/۰۰
۲		۱۵	۰/۲۰۳	۳/۳۰	۰/۲۰۳	۱۸/۷۴	۰/۰۰	۰/۲۰۳	۳/۱۴	۰/۰۰
۳	۲۵	۱۰	۰/۱۳۰	۲/۳۰	۰/۱۳۰	۲۱/۴۶	۰/۰۰	۰/۱۳۰	۱۳/۱۵	۰/۰۰
۴		۱۵	۰/۰۹۰	۲/۱۵	۰/۰۹۰	۲۴/۷۶	۰/۰۰	۰/۰۹۲	۱۳/۸۳	۲/۲۲
۵	۳۰	۱۰	۰/۳۴۳	۱۵۵/۲۵	۰/۳۴۳	۲۶/۱۷	۰/۰۰	۰/۳۴۶	۱۵/۷۶	۰/۸۷
۶		۱۵	۰/۲۱۶	۱۲۰/۵۵	۰/۲۱۶	۲۹/۹۴	۰/۰۰	۰/۲۱۸	۲۱/۰۷	۰/۹۳
۷		۲۰	۰/۱۹۴	۸۰/۸۷	۰/۱۹۴	۳۴/۶۳	۰/۰۰	۰/۱۹۷	۲۷/۳۹	۱/۵۵
۸	۳۵	۱۵	۰/۳۰۵	۳۵۶/۶۲	۰/۳۰۵	۴۵/۹۱	۰/۹۸	۰/۳۱۰	۵۱/۲۴	۱/۶۴
۹		۲۰	۰/۲۰۵	۳۲۲/۶۶	۰/۲۰۹	۵۳/۰۴	۱/۹۵	۰/۲۱۲	۶۳/۷۵	۳/۴۱
۱۰		۲۵	۰/۱۷۸	۶۵۸/۱۲	۰/۱۸۳	۴۸/۲۷	۲/۸۱	۰/۱۸۲	۵۷/۸۴	۲/۲۵
۱۱	۴۰	۱۵	۰/۳۳۹	۶۸۴۵	۰/۳۵۱	۷۹/۷۴	۳/۵۴	۰/۳۴۹	۸۹/۰۹	۲/۹۵
۱۲		۲۰	۰/۲۷۱	۴۷۲۱	۰/۲۷۸	۷۸/۵۸	۲/۵۸	۰/۲۷۹	۹۶/۱۶	۲/۹۵
۱۳		۲۵	۰/۱۸۴	۱۵۶۳	۰/۱۸۸	۸۹/۲۱	۲/۱۷	۰/۱۹۳	۱۰۰/۲۰	۴/۸۹
میانگین			۰/۲۲۷	۱۱۴۱/۱	۰/۲۳۰	۴۳/۴۴	۱/۰۸	۰/۲۳۲	۴۲/۶۶	۱/۸۲

به ترتیب ۱۷۴ ، ۱۵۲ و ۱۲۰ مسئله بصورت بهینه توسط مدل ریاضی، الگوریتم ACS و الگوریتم SA حل شده است. میانگین زمان حل دو الگوریتم نسبت به حل دقیق حدود ۱۱۰۰ ثانیه صرفه‌جویی داشته است. میانگین خطای نسبی الگوریتم ACS برابر ۱/۰۸٪ و الگوریتم SA برابر ۱/۸۲٪ محاسبه شده است. بنابراین علاوه بر اینکه الگوریتم ACS تعداد جواب‌های بهینه بیشتری نسبت به الگوریتم SA بدست آورده است، دارای میانگین خطای کمتری نسبت به الگوریتم SA است. اما میانگین زمان حل الگوریتم SA نسبت به الگوریتم ACS کمتر است.

۵-۲. ابعاد بزرگ

با توجه به اینکه نرم‌افزارGAMS قادر به محاسبه‌ی جواب بهینه‌ی مسائل با ابعاد بزرگ در محدودیت زمانی ۱۰۰۰۰ ثانیه نیست، مسائل با ابعاد بزرگ، تنها با استفاده از الگوریتم‌های ACS و SA

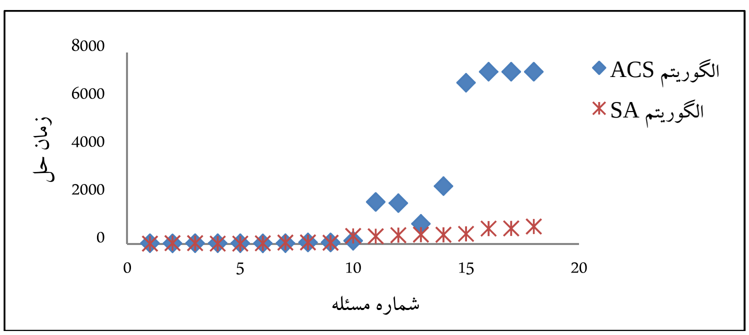
حل شده است و کارایی دو روش نسبت به هم سنجیده می‌شود. برای حل مسائل با روش‌های فراابتکاری محدودیت زمانی ۱۰۰۰۰ ثانیه در نظر گرفته شده است.

طبق روش تولید مسئله که برای ابعاد کوچک تشریح شد، هجده دسته مسئله با ابعاد متفاوت جهت بررسی کارایی روش‌های حل پیشنهادی در حل مسائل با ابعاد بزرگ تولید شده است. در هر دسته، چهارده مسئله‌ی نمونه حل شده و میانگین تابع هدف و میانگین زمان حل مسائل در جدول (۹) گزارش شده است. میانگین کل توابع هدف ۱۸ دسته مسئله با استفاده از الگوریتم ACS نسبت به الگوریتم SA ، ۱/۳٪ کمتر است.

جدول ۹. نتایج حل مسائل تصادفی با ابعاد بزرگ با استفاده از ACS و SA

مشخصات مسئله			الگوریتم ACS		الگوریتم SA	
دسته مسئله	تعداد درس	تعداد استاد	میانگین توابع هدف	میانگین زمان حل (ثانیه)	میانگین توابع هدف	میانگین زمان حل (ثانیه)
۱		۲۵	۰/۱۸۲	۳۵/۱۲	۰/۱۸۱	۱۸/۱۶
۲	۵۰	۳۰	۰/۱۴۵	۲۸/۷۱	۰/۱۴۹	۳۵/۲۷
۳		۳۵	۰/۱۲۵	۳۷/۹۴	۰/۱۲	۳۴/۷۶
۴		۳۰	۰/۱۹۱	۳۹/۴۶	۰/۱۸۹	۱۹/۴
۵	۷۵	۳۵	۰/۱۶۵	۳۶/۶۱	۰/۱۶۳	۱۹/۳۹
۶		۴۰	۰/۱۳۳	۳۲/۷۵	۰/۱۳۵	۳۰/۶۹
۷		۳۵	۰/۱۹۸	۳۷/۲۸	۰/۱۹۹	۶۳/۶۲
۸	۱۰۰	۴۰	۰/۱۶۴	۶۷/۰۸	۰/۱۷۱	۶۸/۹۶
۹		۴۵	۰/۱۷۱	۶۵/۹۸	۰/۱۸۹	۵۴/۳۶
۱۰		۵۰	۰/۲۴۵	۱۳۴/۷۹	۰/۲۵۶	۳۴۱/۱۸
۱۱	۲۰۰	۶۰	۰/۲۲۸	۱۷۶۰	۰/۲۴۶	۳۲۳/۲۹
۱۲		۷۰	۰/۲۱۰	۱۷۱۲/۱۱	۰/۲۱۲	۳۶۹/۴۴
۱۳		۵۰	۰/۲۳۷	۸۴۹/۱۳	۰/۲۲۶	۳۹۴/۷۲
۱۴	۳۰۰	۷۵	۰/۲۸۶	۲۴۱۸	۰/۲۸۸	۳۸۶/۴۷
۱۵		۱۰۰	۰/۲۱۱	۶۷۴۳	۰/۲۲۰	۴۲۴/۸۳
۱۶		۱۰۰	۰/۴۹۲	۱۰۰۰۰	۰/۴۸۴	۴۶۳/۴۴
۱۷	۵۰۰	۱۲۵	۰/۳۷۸	۱۰۰۰۰	۰/۳۸۵	۶۵۲/۰۷
۱۸		۱۵۰	۰/۳۴۱	۱۰۰۰۰	۰/۳۴۸	۷۳۶/۴۸
میانگین			۰/۲۲۸	۲۴۴۴/۳۳	۰/۲۳۱	۲۵۶/۴۷

زمان حل دو الگوریتم ACS و SA در نمودار (۱) نشان داده شده است.



نمودار ۱. مقایسه‌ی زمان حل دو الگوریتم ACS و SA در ابعاد بزرگ

با توجه به نمودار (۱) میانگین زمان حل دو الگوریتم در ابعاد کوچکتر از ۲۰۰ درس مشابه هم است ولی با افزایش ابعاد مسئله زمان حل الگوریتم ACS نسبت به الگوریتم SA افزایش زیادی دارد. در مورد مسائل ۱۶ الی ۱۸ الگوریتم ACS قبل از برقراری شرط توقف به دلیل محدودیت زمانی متوقف می‌شود. افزایش ابعاد مسئله موجب می‌شود که ماتریس فرومون الگوریتم ACS و زمان ایجاد جواب توسط مورچگان افزایش یابد. اما عاملی که افزایش زمان حل را تشدید می‌کند از جستجوی محلی الگوریتم ناشی می‌شود. با توجه به اینکه در الگوریتم جستجوی محلی هر یک از دروس در بین بازه‌های زمانی منتقل می‌شود تا جایی که بهترین جواب حاصل شود، زمان حل ACS با افزایش ابعاد مسئله به شدت افزایش می‌یابد.

۶. نتیجه‌گیری

در این مقاله یک رویکرد دو مرحله‌ای بر اساس برنامه‌ریزی صفر و یک برای مسئله‌ی مطرح شده گسترش داده شده‌است. در مرحله اول تمام دروس اعم از دو واحدی و سه واحدی در روزهای شنبه و یک‌شنبه برنامه‌ریزی می‌شوند. اهداف مدل ریاضی مرحله اول کمینه‌کردن عدم رضایت اساتید از موضوعات درسی و بازه‌های زمانی و همچنین کمینه‌کردن مجموع تعدادتلاقی دروس است. پس از حل مدل ریاضی مرحله اول، برنامه به‌دست آمده برای روزهای دوشنبه و سه‌شنبه تعمیم داده‌شده‌است، بدین معنی که جلسه دوم دروس سه‌واحدی به بازه متناظر در روزهای دوشنبه و یا سه‌شنبه اختصاص داده‌شده‌است. ولی با توجه به اینکه دروس دو واحدی طی یک جلسه در هفته تشکیل می‌شوند. لازم‌است تا حد ممکن بین تمام روزهای هفته توزیع گردند تا بهترین مقدار تابع هدف حاصل شود. بدین منظور مدل ریاضی مرحله دوم به‌کار گرفته می‌شود و تصمیم‌گیری می‌گردد که کدامیک از دروس دو واحدی باید جابجا شود و به کدام بازه زمانی منتقل شود تا بهترین جواب ممکن حاصل شود. با توجه به NP-complete بودن مسائل زمانبندی دروس دانشگاهی استفاده از الگوریتم‌های فراابتکاری زمانی که اندازه مسئله بزرگ باشد، ضروری است. بر این اساس در این مقاله دو الگوریتم فرا ابتکاری سیستم اجتماع مورچگان و شبیه‌سازی تبرید برای حل مسئله ارائه شده‌است. کارایی روش‌های پیشنهادی در ابعاد کوچک از طریق مقایسه با جواب مسئله برنامه‌ریزی صفر و یک و با استفاده از داده‌های دانشکده مهندسی صنایع دانشگاه صنعتی اصفهان و مسائل تصادفی نشان داده‌است. در ابعاد بزرگ کارایی دو روش فرا ابتکاری ارائه شده با استفاده از داده‌های تصادفی تولید شده با هم مقایسه شده است. بر اساس الگوی داده‌های واقعی ۱۸۲ مسئله‌ی تصادفی با ابعاد کوچک و ۲۵۲ مسئله با ابعاد بزرگ تولید شده است. مدل ریاضی قادر به حل بهینه‌ی تمام چهار مسئله مطالعه‌ی موردی با محدودیت زمانی ۱۰۰۰۰ ثانیه شده است. الگوریتم ACS و الگوریتم SA نیز جواب بهینه‌ی به ترتیب ۴ و ۳ مسئله را بدست آورده‌اند. میانگین زمان حل دو الگوریتم فراابتکاری نسبت به حل دقیق حدود ۱۱۰۰ ثانیه صرفه‌جویی داشته است. میانگین خطای نسبی الگوریتم ACS برابر ۱/۰۸٪ و الگوریتم SA برابر ۱/۸۲٪ محاسبه شده است. میانگین کل توابع هدف ۱۸ دسته مسئله‌ی با ابعاد بزرگ با استفاده از الگوریتم ACS نسبت به الگوریتم SA، ۱/۳۲٪ کمتر است. میانگین زمان حل دو الگوریتم در ابعاد کوچکتر از ۲۰۰ درس مشابه هم است ولی با افزایش ابعاد مسئله زمان حل الگوریتم ACS نسبت به الگوریتم SA افزایش زیادی دارد.

پی‌نوشت

1. Educational Timetabling
2. University Course Timetabling Problem
3. Faculty-Course-Timeslot Assignment
4. Analytic Hierarchy Process
5. Analytic Network Process
6. Curriculum Based Course Timetabling
7. Great Deluge algorithm of Dueck
8. Graph Coloring
9. Hacettepe University
10. Memetic Algorithm
11. Binary Integer Programming
12. Directed Multipartite Graph
13. Generalized Traveling Salesman Problem
14. Roulette wheel
15. Epoch

مراجع

[1] Dimopoulou M, Miliotis P. Implementation of a university course and examination timetabling system, European Journal of Operational Research, (2001), Vol. 130, No. 1, pp. 202-213.

[2] Daskalaki S, Birbas T, Housos E. An integer programming formulation for a case study in university timetabling, European Journal of Operational Research, (2004), Vol. 153, No. 1, pp. 117-135.

[3] Daskalaki S, Birbas T. Efficient solutions for a university timetabling problem through integer programming, European Journal of Operational Research, (2005), Vol. 160, No. 1, pp. 106-120.

[4] Al-Yakoob SM, Sherali HD. Mathematical programming models and algorithms for a class - faculty assignment problem, European Journal of Operational Research, (2006), Vol. 173, No. 2, pp. 488-507.

[5] Ismayilova NA, Sagir M, Gasimov RN. A multiobjective faculty-course-time slot assignment problem with preferences, Mathematical and Computer Modelling, (2007), Vol. 46, No. 7-8, pp. 1017-1029.

[6] Burke EK, Marecek J, Parkes AJ, Rudova H. A branch-and-cut procedure for the udine course timetabling problem, Annals of Operations Research, (2012), pp. 1-17.

[7] Lach G, Lubbecke ME. Curriculum based course timetabling: new solutions to Udine benchmark instances, Annals of Operations Research, (2012), Vol. 194, No. 1, pp. 255-272.

[8] Hao JK, Benlic U. Lower bounds for the ITC-2007 curriculum-based course timetabling problem, European Journal of Operational Research, (2011), Vol. 212, No. 3, pp. 464-472.

[9] Lewis R. A survey of metaheuristic-based techniques for university timetabling problems, OR Spectrum, (2008), Vol. 30, No. 1, pp. 167-190.

[10] Socha K, Knowles J, Sampels M. A max-min ant system for the university course timetabling problem, Ant Algorithms, (2002), pp. 63-77.

[11] Burke E, Bykov Y, Newall J, Petrovic S. A time - predefined approach to course timetabling, The Yugoslav Journal of Operations Research, (2003), Vol. 13, No. 2.

[12] Kostuch P. The university course timetabling problem with a three-phase approach, Practice and Theory of Automated Timetabling V, (2005), pp. 109-125.

[13] Chiarandini M, Birattari M, Socha K, Rossi - Doria O. An effective hybrid algorithm for university course timetabling, Journal of Scheduling, (2006), Vol. 9, No. 5, pp. 403-432.

[14] Aladag CH, Hocaoglu G, Basaran MA. The effect of neighborhood structures on tabu search algorithm in solving course timetabling problem, Expert Systems with Applications, (2009), Vol. 36, No. 10, pp. 12349-12356.

[15] Lu Z, Hao JK. Adaptive tabu search for course timetabling, European Journal of Operational Research, (2010), Vol. 200, No. 1, pp. 235-244.

[16] Shiau DF. A hybrid particle swarm optimization for a university course scheduling problem with flexible preferences, Expert Systems with Applications, (2011), Vol. 38, No. 1, pp. 235-248.

[17] Abdullah S, Turabieh H. On the use of multi neighborhood structures within a tabu-based memetic approach to university timetabling problems, Information Sciences, (2012), pp. 146-168.

[18] Muller B, Reinhardt J, Strickland MT. Combinatorial optimization, Neural Networks, (1995), pp. 126-134.

[19] Taghavifard M, Sheikh K, Shahsavari A. Modified ant colony algorithm for the vehicle routing problem with time windows, IJIEPM, (2009), Vol. 20, No. 2, pp. 23-30.

[20] Radmehr F, Shams Gharneh N. Forecasting Tehran's bourse price index (TEPIX) using high-order fuzzy time series and simulated annealing algorithm, IJIEPM, (2013), Vol. 24, No. 1, pp. 95-106.

[21] Seifi M, Tavakkoli-Moghaddam R, Jolai F. The use of simulated annealing and genetic algorithms for a multi - mode resource constrained project scheduling problem with discounted cash flows, IJIEPM, (2008), Vol. 19, No. 4, pp. 85-91.

[22] Taleizadeh A, Aryanezhad M, Makoee A. Utilization of simulated annealing in optimization of a multi-product inventory control model with stochastic replenishment and space constraint, IJIEPM, (2009), Vol. 20, No. 2, pp. 1-10.