

حل مسایل زمانبندی پروژه‌ها با منابع محدود با استفاده از الگوریتم مورچگان اصلاح شده

کاوه خلیلی دامغانی¹، رضا توکلی مقدم^{2*} و مجتبی طبری³

¹ استادیار گروه مهندسی صنایع - دانشگاه آزاد اسلامی - واحد علوم و تحقیقات

² استاد گروه مهندسی صنایع - پردیس دانشکده‌های فنی - دانشگاه تهران

³ استادیار گروه مدیریت دولتی - واحد قائم‌شهر - دانشگاه آزاد اسلامی - قائم‌شهر

(تاریخ دریافت 85/6/27، تاریخ دریافت روایت اصلاح شده 89/12/18، تاریخ تصویب 90/1/30)

چکیده

موضوع زمانبندی پروژه‌ها با منابع محدود¹ (RCPSP) در پی یافتن توالی مناسبی برای انجام فعالیت‌های یک پروژه است؛ به نحوی که محدودیت‌های تقدم و و تأخر شبکه پروژه و انواع مختلف محدودیت‌های منبعی موجود در پروژه به طور همزمان ارضا و معیار سنجش معینی از جمله زمان انجام پروژه، هزینه انجام، تعداد فعالیت‌های تأخیردار و غیره بهینه شوند. RCPSP، یک مسئله چندجمله‌ای غیرقطعی سخت² به شمار می‌آید و اهمیت این موضوع در ابعاد عملی باعث شده است که تا کنون رویکردهای فراابتکاری متعددی برای حل این موضوع ارائه شود. در این مقاله رویکردی بر اساس بهینه‌سازی توسط کلونی مورچگان برای حل مسئله زمانبندی پروژه‌ها با منابع محدود ارائه شده است. از جمله تفاوت‌های اصلی رویکرد ارائه شده در این مقاله می‌توان به تعریف قانون انتخاب احتمالی به شکل نوین، تغییر عوامل الگوریتم به شکل تطبیقی، جلوگیری از بروز رفتارهای نامناسب و تعیین رفتار کلی الگوریتم در تکرارهای بالا اشاره کرد. در مورد نبود قطعیت برخی از عوامل مسئله نیز بحث و بررسی شده است. الگوریتم با استفاده از نرم افزار VB 6.0 کد شده و بر مثال‌های الگو³ در این زمینه آزمایش شده است. نتایج حاصل امیدوارکننده بوده و با جواب‌های بهینه در صورت وجود یا با بهترین جواب‌های یافت شده مقایسه شده‌اند.

واژه‌های کلیدی: زمانبندی پروژه با منابع محدود، روش‌های فراابتکاری، بهینه‌سازی توسط کلونی مورچگان

مقدمه

مثال‌های الگو آزمایش می‌کنند [8]. این مثال‌ها توسط نرم‌افزار PROGEN یا PROGEN/MAX در شرایط کنترل شده‌ای تولید شده‌اند. در واقع می‌توان گفت این مثال‌ها در سطوح مختلفی از پیچیدگی تولید شده‌اند و الگوریتم‌های جدید در صورت ارائه جواب مناسب و قابل قبول برای این مثال‌ها کارا شمرده می‌شوند [8].

از جمله رویکردهای موفق برای حل این مسئله می‌توان بهینه‌سازی توسط کلونی مورچگان را نام برد. در سال‌های اخیر محققان با استفاده از این رویکرد به موفقیت‌های زیادی در مورد حل سایر مسایل بهینه‌سازی ترکیبی نیز دست یافته‌اند [16-9,20]. منشاء پیدایش این دسته از رویکردهای حل (بهینه‌سازی توسط کلونی مورچگان) یکی از ویژگی‌های پیچیده حشراتی است که به طور اجتماعی زندگی می‌کنند. این ویژگی، استیگمرجی نام دارد. استیگمرجی دسته‌ای از مکانیزم‌های ایجاد ارتباطات متقابل میان حیوانات است [13]. با استفاده از استیگمرجی، حشرات قادرند کوتاه‌ترین مسیرهای میان

موضوع زمانبندی پروژه‌ها با منابع محدود (RCPSP) در واقع کلی‌ترین موضوع زمانبندی است که مسایل زمانبندی کار کارگاهی، زمانبندی جریان کارگاهی و سایر مسایل زمانبندی همگی زیر مجموعه‌ای از این موضوع به حساب می‌آیند [1 و 2]. به طور کلی، RCPSP در پی یافتن توالی مناسبی برای انجام فعالیت‌های یک پروژه است، به نحوی که محدودیت‌های تقدم و تأخر شبکه پروژه و انواع مختلف محدودیت‌های منبعی موجود در پروژه به طور همزمان ارضا شوند و معیار سنجش معینی از جمله زمان انجام پروژه، هزینه انجام، تعداد فعالیت‌های تأخیردار و غیره بهینه شوند [3]. مدل‌های مختلفی تا کنون برای حل این مسئله توسعه داده شده‌اند [4 و 5]. همچنین این مسئله از ابعاد عملی و علمی بسیار پر اهمیت است [6]. این مسئله به چندین شاخه مختلف تقسیم‌بندی شده است [7]. برای هر یک از انواع RCPSP، دسته‌ای از مثال‌های الگو تولید شده‌اند که محققان الگوریتم‌ها و رویکردهای حل خود را با استفاده از این

مثال‌های الگوی انتخاب شده، تنظیمات عوامل الگوریتم، نتایج عددی و پیشنهادهای ارائه می‌شوند.

مدل برنامه‌ریزی ریاضی شرح مسئله

برای زمانبندی یک پروژه با منابع محدود با حد بالای اجرای پروژه، در حالتی که همه عوامل مدل قطعی هستند و برای انجام و تکمیل پروژه باید تعدادی فعالیت انجام شوند. در هر بازه زمانی و نیز در کل پروژه تعدادی منبع تجدیدپذیر، منبع تجدیدناپذیر و غیره در اختیار داریم. انواع منابعی که در اختیار داریم نیز محدود است. هر فعالیت را می‌توان فقط به یک شیوه انجام داد که زمان انجام فعالیت‌ها و منابع مورد نیاز آن فعالیت با سایر فعالیت‌ها فرق می‌کند و مقداری قطعی است. تابع هدف مدل عبارت از کمینه کردن زمان انجام کل پروژه است، به نحوی که روابط تقدم و تأخر میان فعالیت‌ها رعایت شود و محدودیت‌هایی که در مورد منابع وجود دارند نیز نقض نشوند. به محض شروع یک فعالیت، امکان توقف در انجام آن وجود ندارد. چندین فعالیت که نیاز به یک نوع منبع مشترک دارند، می‌توانند در یک زمان با همدیگر انجام شوند؛ فقط و فقط به شرطی که محدودیت‌های منابع را در هر دوره نقض نکنند. مدل برنامه‌ریزی ریاضی زیر برای حل بهینه یک چنین مسئله‌ای توصیه می‌شود. عوامل اصلی این مدل به شرح زیر هستند:

تعریف علائم و عوامل

J : تعداد فعالیت‌های پروژه است

d_j : مدت زمانی که به طول می‌انجامد تا فعالیت j انجام شود

$R(N, D)$: مجموعه همه منابع تجدیدپذیر و تجدیدناپذیر

\bar{T} : حد بالایی برای انجام و تکمیل کل پروژه

$K_n^p \geq 0$: تعداد واحدهایی که از منبع تجدیدپذیر نوع r در دوره t ($t=1, \dots, \bar{T}$) در دسترس است
 $r \in R (r \in D)$

$K_r^v \geq 0$: جمع کل واحدهای موجود از منابع تجدیدناپذیر نوع r , $r \in N (r \in D)$

لانه و غذا را بیابند و یا لانه بسازند و لانه را مرتب کنند. طبق واژه استیگمرجی یک علامت در محیط باعث انجام یک عمل در حشرات می‌شود. حشره به محیط پیرامون خود می‌نگرد و در اثر علائمی که در محیط می‌یابد و شرایط فعلی محیط برای انجام یک فعالیت خاص انگیزه پیدا می‌کند. برای ایجاد تغییر و تحول در محیط پیرامون، دو نظریه کلی وجود دارد که هر یک منجر به ایجاد رفتار خاصی از سوی حشرات می‌شود. در نوع خاصی از استیگمرجی یک مکانیزم هشداردهنده (یا یک علامت) برای هماهنگ کردن حشرات مورد استفاده قرار می‌گیرد. در واقع در این نوع استیگمرجی، حشرات تغییر فیزیکی بر محیط اعمال نمی‌کنند، بلکه با استفاده از علائمی که از خود باقی می‌گذارند (که اغلب مواد خاصی است) با هم هماهنگ می‌شوند و ارتباط برقرار می‌کنند. برای مثال مورچه‌ها از خود ماده شیمیایی بوداری به نام فرمون باقی می‌گذارند و میزان چگالی فرمون روی مسیری که مورچه‌ها از آن عبور کرده‌اند در هماهنگ کردن آنها برای انجام اعمال خاصی مؤثر است. این رفتار فرمون‌گذاری مورچه‌ها در هماهنگ کردن خودشان با هم نوعی خودسازماندهی به حساب می‌آید و مورچه‌ها را در امور مختلفی که مهم‌ترین آنها یافتن کوتاه‌ترین مسیر از لانه به غذا و بر عکس است، یاری می‌دهد. این رفتار مورچه‌ها منشا پیدایش الگوریتم‌های موفق برای حل مسایل بهینه‌سازی ترکیبی به نام الگوریتم‌های مورچگان شده است.

در این مقاله، رویکردی بر مبنای ACO برای حل RCPSP ارائه می‌شود. الگوریتم مورچگان مورد نظر از ردهای فرمون و اطلاعات ابتکاری به طور همزمان برای انتخاب و زمانبندی فعالیت‌ها بهره می‌برد. اطلاعات ابتکاری نرمالیزه شده و جدید هستند و به صورت پویا محاسبه می‌شوند. عوامل الگوریتم به طور پویا تنظیم می‌شوند و خصوصیات دیگری که در قسمت‌های مختلف این الگوریتم گنجانده شده، باعث شده است که در مجموع نتایج بهتری نسبت به سایر رویکردهای حل، برای این مسئله ارائه کند. در قسمت‌های بعدی این مقاله به ترتیب مدل برنامه‌ریزی ریاضی پیشنهادی، عدم قطعیت در عوامل مدل ریاضی و رویکرد حل مدل برنامه‌ریزی ریاضی در حالت احتمالی [17-19]، الگوریتم مورچگان برای حل RCPSP، ویژگی‌های جدید گنجانده شده در الگوریتم،

مدل ریاضی زیر یک مدل برنامه‌ریزی صفر و یک است؛ به طوری که X_{jt} یک متغیر صفر - یک با تعریف زیر بوده و هدف آن کمینه‌کردن زمان اجرای کل پروژه است:

$$t = EF_j, \dots, \bar{T}$$

$$X_{jt} = \begin{cases} 1 & \text{if activity } j \text{ is finished in period } t, \\ 0 & \text{otherwise.} \end{cases}$$

نبود قطعیت عوامل مسئله

در حالت احتمالی فرض می‌شود که تخمین دقیقی از زمان انجام فعالیت‌ها و حداکثر مقدار منابع در دسترس وجود ندارد. بنابراین چنین مسئله‌ای را می‌توان به شکل برنامه‌ریزی ریاضی نشان داد. در این مدل، از مفهوم سطح اطمینان برای احتمالی کردن مدل استفاده شده است. در کل سه نوع سطح اطمینان برای مدل تعریف شده است که به ترتیب مربوط به زمان‌های انجام فعالیت‌ها، مقدار منبع تجدیدپذیر در دسترس و مقدار منبع تجدیدناپذیر در دسترس هستند.

$k_{jr}^p \geq 0$: تعداد واحدهای مورد نیاز از منبع تجدیدپذیر نوع r که توسط فعالیت j به ازای هر دوره که فعالیت انجام می‌گیرد، مصرف می‌شود ($r \in R (r \in D)$)
 $k_{jr}^v \geq 0$: تعداد واحدهای مورد نیاز از منابع تجدیدناپذیر نوع r که توسط فعالیت j مصرف می‌شود
 $P_j (S_j)$: مجموعه پیش‌نیازها (پس‌نیازهای) بدون وقفه فعالیت j

EF_j : زودترین زمان پایان فعالیت j ، که با توجه به حداقل مدت زمان‌های فعالیت‌ها و بدون در نظر گرفتن مقدار منبع مصرفی و محدودیت منابع محاسبه می‌شود (خروجی نرم افزارهای کنترل پروژه).
 LF_j : دیرترین زمان پایان فعالیت j ، که با توجه به حداقل مدت زمان‌های فعالیت‌ها و بدون در نظر گرفتن مقدار منبع مصرفی و محدودیت منابع و با احتساب حد بالای \bar{T} که برای کل زمان پروژه در نظر گرفته شده است، محاسبه می‌شود (خروجی نرم افزارهای کنترل پروژه)

$$\min \Phi(x) = \sum_{t=EF_j}^{\bar{T}} t \cdot x_{jt} \quad (1)$$

s.t.

$$\sum_{t=EF_j}^{\bar{T}} x_{jt} = 1 \quad j = 1, \dots, J \quad (2)$$

$$\sum_{t=EF_h}^{\bar{T}} t \cdot x_{ht} \leq \sum_{t=EF_j}^{\bar{T}} (t - d_j) x_{jt} \quad j = 2, \dots, J, h \in p_j \quad (3)$$

$$\sum_{j=1}^J k_{jr}^p \sum_{q=\max\{t, EF_j\}}^{\min\{t+d_j-1, \bar{T}\}} x_{jq} \leq K_{rt}^p \quad r \in R, t = 1, \dots, \bar{T} \quad (4)$$

$$\sum_{j=1}^J k_{jr}^v \sum_{t=EF_j}^{\bar{T}} x_{jt} \leq K_r^v \quad r \in N \quad (5)$$

$$x_{jt} \in \{0, 1\} \quad j = 1, \dots, J, t = 1, \dots, \bar{T} \quad (6)$$

که کل پروژه در سطح اطمینان تک تک فعالیت‌ها قابل تکمیل است. در واقع در اینجا پروژه یک سیستم سری از دید قابلیت اطمینانی است که اجزای آن فعالیت‌هایی هستند و احتمال کارکرد هر جزء همان سطح اطمینانی است که مدیریت آن را تعیین می‌کند و پایایی سیستم عبارت از احتمال کارکرد تمام اجزای سیستم و یا احتمال تکمیل همه فعالیت‌های پروژه در زمانی حداکثر معادل با حداکثر زمان محاسبه شده برای آنها است.

طبق مطالب ذکر شده فقط اطلاعاتی که از مقدار در دسترس منابع تجدیدپذیر وجود دارد، دو حد بالا و پایین برای مقادیر این منابع است و منابع می‌توانند هر مقداری را با احتمال مساوی در بین این دو حد داشته باشند. بنابراین می‌توان فرض کرد که این منابع تابع چگالی یکنواخت در بازه مورد نظر هستند:

$$K_{rt}^{*\rho} \sim U(K_{rt_{MIN}}^{\rho}, K_{rt_{MAX}}^{\rho}) \quad (10)$$

طبق مطالبی که در بخش‌های قبلی عنوان شد، اگر مایل باشیم که محدودیت مربوط به مقدار منابع تجدیدپذیر در 95 درصد مواقع ارضا شود، آنگاه داریم:

$$K_{jr}^{*\rho} = r_j^* \geq 0 \quad (11)$$

$$\int_{-\infty}^{r_j^*} r_j(t_j) dt_j = 1 - \alpha_{renewable} \quad (12)$$

$$\int_{-\infty}^{K_{rt}^{*\rho}} \frac{K_{rt}^{\rho} - K_{rt_{MIN}}^{\rho}}{K_{rt_{MAX}}^{\rho} - K_{rt_{MIN}}^{\rho}} dt_j = 0.95 \quad (13)$$

در مورد منابع تجدیدناپذیر هم به همین ترتیب عمل می‌کنیم. مدل (14)-(19) معادل احتمالی مدل (1)-(6) است.

در قسمت بعد به شرح دقیق الگوریتم و ضمائم آن و آزمایش آن روی مثال‌های الگو که به طور تصادفی و در درجه سختی معین و توسط نرم‌افزار PROGEN تولید شده‌اند، پرداخته می‌شود [8]. پس از اینکه معادل قطعی مدل برنامه‌ریزی ریاضی احتمالی توسعه داده شد، می‌توان برای حل آن از الگوریتم مورچگانی که در زیر معرفی می‌شود، استفاده کرد:

در این مقاله، فرض می‌شود عوامل تصادفی توزیع‌های شناخته شده دارند. چنین مدلی را در ادبیات برنامه‌ریزی احتمالی، برنامه‌ریزی شانس محدودیت دار⁴ می‌نامند، اگر سطوح اطمینان تعریف شده برای هر یک از محدودیت‌ها با بقیه متفاوت باشد، آن را برنامه‌ریزی شانس محدودیت‌دار منفرد می‌نامند و در صورتی که برای دسته‌ای از محدودیت‌ها، یک سطح اطمینان در نظر گرفته شود، مدل برنامه‌ریزی شانس محدودیت دارهمزمان⁵ نام خواهد داشت [18].

اگر فرض کنیم که زمان‌های انجام فعالیت‌ها از توزیع نرمال با میانگین و واریانس مشخصی پیروی می‌کنند، یعنی $d_j \sim N(\mu_j, \sigma_j^2)$; $J = 1, 2, \dots, j$ و مقدار منابع در دسترس تجدیدپذیر توزیع یکنواخت در بازه $[K_{rt_{MIN}}^{\rho}, K_{rt_{MAX}}^{\rho}]$ و مقدار منابع تجدیدناپذیر نیز توزیع یکنواخت در بازه $[K_{rt_{MIN}}^{\nu}, K_{rt_{MAX}}^{\nu}]$ دارد، محاسبات لازم برای تبدیل مدل احتمالی به مدل قطعی به این ترتیب است:

$$d_j \sim N(\mu_j, \sigma_j^2) ; J = 1, 2, \dots, j \quad (7)$$

باشیم که محدودیت مربوط به زمان انجام فعالیت‌ها در 95 درصد مواقع ارضا شود، آنگاه داریم:

$$\int_{-\infty}^{d_j^*} f_j(d_j) dt_j = 1 - \alpha_{dj} \quad (8)$$

$$\int_{-\infty}^{d_j^*} \frac{1}{\sigma_j \sqrt{2\pi}} e^{-\frac{(d_j - \mu_j)^2}{2\sigma_j^2}} dt_j = 0.95 \quad (9)$$

با قرار دادن مقدار قطعی d_j^* در 95 درصد مواقع مطمئن هستیم که محدودیت ذکر شده ارضا خواهد شد. اما در مورد کل پروژه این مطلب صحیح نیست. زیرا برای پایان یک پروژه باید همه فعالیت‌های آن انجام شوند. این فعالیت‌ها در زمانبندی به دنبال یکدیگر مرتب می‌شوند و به طور دقیق حالت یک سیستم سری را به کل مجموعه خواهند داد. به طور روشن‌تر اگر فرض کنیم برای زمان انجام یک فعالیت، سطح اطمینان مشخصی تعریف شد و در آن سطح اطمینان، حداکثر زمان انجام آن فعالیت به دست آمد تا محدودیت‌های احتمالی را ارضا کند و این کار برای تک تک فعالیت‌های پروژه انجام شد، نمی‌توان گفت

$$\min \Phi(x) = \sum_{t=EF_j}^{\bar{T}} t \cdot x_{jt} \quad (14)$$

s.t.

$$\sum_{t=EF_j}^{\bar{T}} x_{jt} = 1 \quad j=1, \dots, J \quad (15)$$

$$\Pr\left[\sum_{t=EF_h}^{\bar{T}} t \cdot x_{ht} \leq \sum_{t=EF_j}^{\bar{T}} (t-d_j)x_{jt}\right] \geq (1-\alpha_{dj}) \quad j=2, \dots, J, h \in p_j \quad (16)$$

$$\Pr\left[\sum_{j=1}^J k_{jr}^p \sum_{q=\max\{t, EF_j\}}^{\min\{t+d_j-1, \bar{T}\}} x_{jq} \leq K_{rt}^p\right] \geq (1-\alpha_{renewable}) \quad r \in R, t=1, \dots, \bar{T} \quad (17)$$

$$\Pr\left[\sum_{j=1}^J k_{jr}^v \sum_{t=EF_j}^{\bar{T}} x_{jt} \leq K_r^v\right] \geq (1-\alpha_{non-renewable}) \quad r \in N \quad (18)$$

$$x_{jt} \in \{0,1\} \quad j=1, \dots, J, t=1, \dots, \bar{T} \quad (19)$$

نشان می‌دهد و در واقع نشان‌دهنده انجام همین تخصیص توسط مورچه‌های قبلی است. η_{ij} : (اطلاعات ابتکاری مطلق) بر تمایل زمانبندی فعالیت j ام در موقعیت i ام توالی فعالیت‌ها دلالت دارد. این تمایل با استفاده از قوانین ابتکاری و ساختار مسئله قابل تعریف است.

p_{ij}^k : احتمال انتخاب فعالیت j ام توسط k امین مورچه برای قرارگیری در موقعیت i ام توالی فعالیت‌ها. α, β : عواملی هستند که اهمیت نسبی ردپای فرومون و اطلاعات ابتکاری را برای انتخاب شهر بعدی نشان می‌دهند.

مرحله 3: تعریف مورچه‌های مصنوعی هوشمند با قدرت یادگیری

الف) تعریف ردهای فرومون به عنوان قدرت یادگیری مورچه‌ها

ردهای فرومون به شکل یک ماتریس به الگوریتم وارد می‌شوند که در این ماتریس عنصر j, i میزان رد فرومونی را نشان می‌دهد که بیانگر زمانبندی فعالیت j ام در موقعیت i ام توالی است. مقادیر بزرگ‌تر نشان می‌دهد که فعالیت j ام دفعات بیشتری در موقعیت i ام توالی زمانبندی شده است.

ب) تعریف اطلاعات ابتکاری: (اطلاعات ابتکاری مطلق)

اطلاعات ابتکاری را اینگونه تعریف کنید:

الگوریتم مورچگان برای حل RCPSP

مرحله 1: محاسبات لازم قبل از شروع الگوریتم:

الف) تشکیل یک گراف فعالیت روی گره⁶ برای پروژه. ب) محاسبات مسیر برگشت و تعیین دیرترین و زودترین زمان پایان فعالیت‌ها (از نرم‌افزارهای کنترل پروژه بهره بگیرید). ج) انتخاب طرح تولید زمانبندی سری و تهیه یک فهرست از فعالیت‌های مجاز منبعی-اولویتی برای هر مرحله از زمانبندی.

مرحله 2: ساختن جواب

قانون انتخاب احتمالی اصلاح شده

مورچه‌ها ساخت جواب را با شروع از گره (فعالیت) صفر آغاز می‌کنند و آن را به موقعیت صفر اختصاص می‌دهند. پس از آن در هر مرحله با استفاده از رابطه زیر گره (فعالیت) بعدی را انتخاب می‌کنند:

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta} \quad \text{if } j \in N_i^k \quad (20)$$

N_i^k : مجموعه فعالیت‌های مجاز برای زمانبندی در هر مرحله‌اند که عبارت از فعالیت‌هایی هستند که هم محدودیت‌های منبعی و هم محدودیت‌های اولویتی را ارضا می‌کنند.

τ_{ij} : (رد فرومون مطلق): مقداری است که تمایل زمانبندی فعالیت j ام در موقعیت i ام توالی فعالیت‌ها

الف) تعداد تکرارها

ب) مقایسه با جواب بهینه در صورت بودن

شکل (1) شبه کد الگوریتم مورچگان برای حل RCPSP را نشان می‌دهد.

نتایج محاسباتی

مثال‌های الگو

برای آزمایش کردن مدل برنامه‌ریزی صفر و یک که در قسمت‌های قبلی ارایه شد، از سری مثال‌های کتابخانه مسایل الگو که دارای 30 فعالیت، 4 نوع منبع تجدیدپذیر بود، مثال نمونه J301_1 و J309_1 انتخاب شد. جواب‌ها با بهترین جواب‌های موجود در کتابخانه مسایل الگو مقایسه شد. همان طور که در جدول (1) نشان داده شده است، منظور از parameter در ستون اول جدول (1) تنظیمات نرم افزار ProGen است.

جدول 1: بهترین جواب شناخته شده پروژه‌های 30 فعالیته

Parameter	Instance	Makespan	CPU-Time (sec)
1	1	43	0.3
9	1	83	423.46

نرم افزار و سخت‌افزار

الگوریتم مورچگان تشریح شده در فصل قبل با استفاده از نرم افزار VB 6.0 کد شد. مثال J301_1 و مثال J309_1 با استفاده از این کد با شرایط کنترل شده‌ای حل شد. کد نوشته شده روی کامپیوتری با مشخصات Windows XP professional و Pentium III 500MHz و V.2002 و 128M.RAM آزمایش شد.

آزمایش‌های طراحی شده

برای بررسی رفتار الگوریتم طراحی شده و مقایسه آن با روش‌های قبلی، طراحی الگوریتم‌هایی از این دست و همچنین تعیین تأثیر تنظیمات مناسب عوامل بر رفتار کلی الگوریتم، سه آزمایش به شکل زیر طراحی شد که از نظر نوع اطلاعات ابتکاری، تعداد نسل‌های مورچه‌ها، نرخ تبخیر فرمون و تأثیر ردهای فرمون و اطلاعات ابتکاری در انتخاب احتمالی با هم متفاوت هستند. ویژگی‌های این آزمایش‌ها در جدول (2) خلاصه شده‌اند. در قسمت بعد نتایج این آزمایش‌ها بر مثال‌های الگو ارایه شده‌اند. همه نتایج میانگین 10 بار اجرای الگوریتم هستند.

$$\eta_{ij} = \text{Max}_{l \in N} LS_l - LS_j + 1 \quad (21)$$

LS_j : دیرترین زمان شروع فعالیت j ام به طوری که به حد بالای زمان اجرای پروژه طبق محدودیت‌ها خدشه وارد نشود. این مقدار از زمانبندی پروژه بدون در نظر گرفتن محدودیت منابع به دست می‌آید.
 N_i^k : مجموعه فعالیت‌هایی که در هر لحظه از ساخت جواب نیمه کاره، هم از نظر تقدم و تأخر و هم از نظر منبعی شدنی هستند (زیر مجموعه‌ای از مجموعه پیش‌نیازها و پس‌نیازها که از نظر منبعی نیز شدنی هستند).

اطلاعات ابتکاری به فرم یک ماتریس و به صورت مطلق به الگوریتم وارد می‌شوند. منظور از اطلاعات ابتکاری مطلق، میزان تمایل قرار دادن فعالیت j ام در موقعیت i ام توالی فعالیت‌ها است.

مرحله 4: به روزآوری فرمون

الف) تبخیر فرمون (تبخیر سراسری)

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} \quad (22)$$

ρ : نرخ تبخیر فرمون ($0 < \rho < 1$)

تبخیر سراسری فرمون در این الگوریتم قبل از باقی گذاردن فرمون انجام می‌شود.

ب) باقی گذاردن فرمون (سراسری): منظور از باقی گذاردن فرمون به روش سراسری این است که بعد از تکمیل جواب، مورچه از خود روی مسیر فرمون می‌ریزد:

$$\tau_{ij} \leftarrow \tau_{ij} + \rho \cdot \Delta \tau_{ij}^k \quad (23)$$

$$\Delta \tau_{ij}^k = \begin{cases} (2T)^{-1} & \text{if activity } j \text{ is scheduled in} \\ & \text{position } i \text{ of sequence by ant } k, \\ 0 & \text{otherwise.} \end{cases}$$

$\Delta \tau_{ij}^k$: مقدار فرمونی که بعد از ساخته شدن یک جواب توسط مورچه k ام به گره‌های مسیر عبوری اضافه می‌شود.

T : عبارت از زمان انجام کل پروژه توسط مورچه k ام است.

مرحله 5: تعریف شرایط توقف و چاپ کردن بهترین جواب

بررسی کردن یکی از شرایط توقف زیر:

تشخیص داده است و آنها را از طریق ردهای فرامون ریخته شده روی آنها به طور دقیق شناسایی کرده است، در ابتدای مرحله بهره برداری⁷ اثر اطلاعات ابتکاری در انتخاب احتمالی به کلی از بین می‌رود و مورچه‌ها فقط طبق آنچه از تکرارهای قبلی آموخته‌اند عمل می‌کنند. بنابراین بار محاسباتی الگوریتم کاسته خواهد شد. الگوریتم 1000 تکرار در این وضعیت باقی خواهد ماند. اما در این 1000 تکرار نیز برای جلوگیری از بروز رفتار ایستا، نرخ تبخیر فرامون در 200 تکرار آخر افزایش می‌یابد تا مورچه‌ها فرایند جستجوی خود را در اطراف بهترین جواب‌های یافته شده تا آن زمان متمرکز کنند و جستجوی نهایی خود را در همان نواحی از فضای حل به پایان برسانند.

اطلاعات ابتکاری به تعداد مراحل زمانبندی و برای لیست شدنی و از نوع nLS محاسبه شدند، در هر یک از مراحل انتخاب الگوریتم لیست شدنی محاسبه شد. قانون انتخاب احتمالی ساده بود و کل تکرارهای الگوریتم برابر با 2000 در نظر گرفته شد. اگر الگوریتم برای بیش از 10 درصد از کل تکرارهای خود یک جواب تکراری را چندین بار تولید کند، نشان‌دهنده ضعف تنظیمات پارامتری و رسیدن به حالت ایستا است. شکل‌های (2) الی (6) بعد از نتایج آزمایش‌ها و انحراف از بهینگی برای هر آزمایش و بر دو مثال انتخاب شده از مثال‌های الگو را نشان می‌دهد. همچنین اثر تنظیمات مناسب عوامل بر رفتار الگوریتم به خوبی مشخص است.

```

procedure ACO for RCPS
  ReadInstance
  ScheduleUnconstrainedProject
  InitializeParameters & Ants
  while(TerminationConditionNotMet) Do
    GenerateSolution
    ComputeResourcePrecedenceEligibleNeighborhoodLists
    TimeAddition
    ComputeDynamicHeuristicInformation
    ComputeProbabilisticValue
    SelectSequence
    UpdateResource
    UpdateStatistics
    UpdatePheromoneTrails
    SavingSolutionInAntMemory
  end while
  printBestSolutionFoundInAllGenerations
end procedure

```

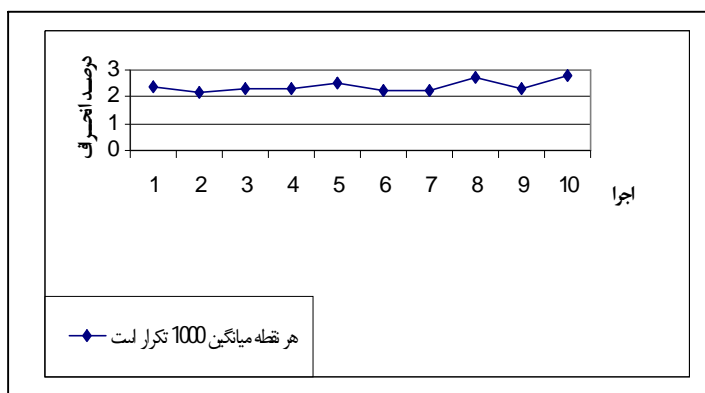
شکل 1: شبه کد الگوریتم مورچگان برای حل RCPS

تحلیل نتایج آزمایش‌ها

در آزمایش نهایی (سوم)، نرخ تبخیر، متغیر و دو مرحله‌ای بود تا در مراحل پایانی اجرا، الگوریتم دچار رفتار ایستا نشود. در مرحله اول نرخ تبخیر فرامون برابر 0/025 و در مرحله دوم برابر 0/075 بود. اثر ردهای فرامون روی انتخاب‌های بعدی، تأثیر اطلاعات ابتکاری روی انتخاب‌های بعدی متغیر بود و با هر تکرار از الگوریتم به طور خطی کاهش می‌یافت تا بعد از نیمی از کل تکرارهای الگوریتم به عدد صفر می‌رسید. تغییر میزان اثر اطلاعات ابتکاری در احتمال انتخاب باعث جلوگیری از بروز رفتار ایستا خواهد بود، زیرا بعد از گذشت 1000 تکرار در هر اجرا که الگوریتم مسیرهای مناسب را به طور کامل یافته و

جدول 2: ویژگی‌های آزمایش‌ها

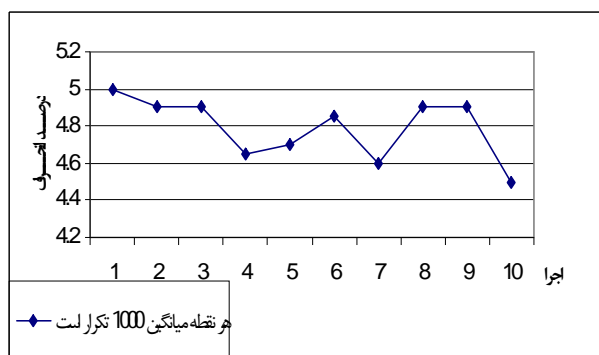
مورچه	تکرار	تبخیر	α	β	به روزآوری	نوع	ابتکاری	لیست شدنی	انتخاب	نام
1	1000	1 نرخ	1	2	سراسری	ثابت	LS	دارد	احتمالی	آزمایش 1
1	2000	1 نرخ	1	1	سراسری	ثابت	LS	دارد	احتمالی	آزمایش 2
1	2000	2 نرخ	1	خطی	سراسری	پویا	nLS	دارد	احتمالی	آزمایش 3



شکل 2: انحراف از بهینه: مثال J301_1، آزمایش 1

جدول 3: نتایج آزمایش‌ها روی مثال J301_1

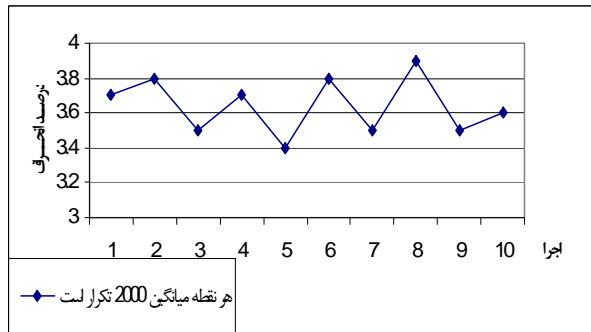
نام	OPT.	F.S.	B.S. (max)	B.S. Time (sec)	I.T. (sec)	S.B.	Deviation
آزمایش 1	مشاهده نشد	اولین تکرار	50 تکرار	15	510	ندارد	4.8
آزمایش 2	مشاهده نشد	اولین تکرار	50 تکرار	15	617	ندارد	3.6
آزمایش 3	مشاهده نشد	اولین تکرار	50 تکرار	15	1915	ندارد	2.3



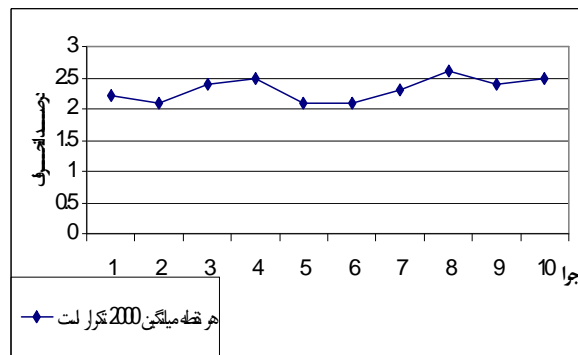
شکل 3: انحراف از بهینه: مثال J309_1، آزمایش 1

جدول 4: نتایج آزمایش‌ها روی مثال J309_1

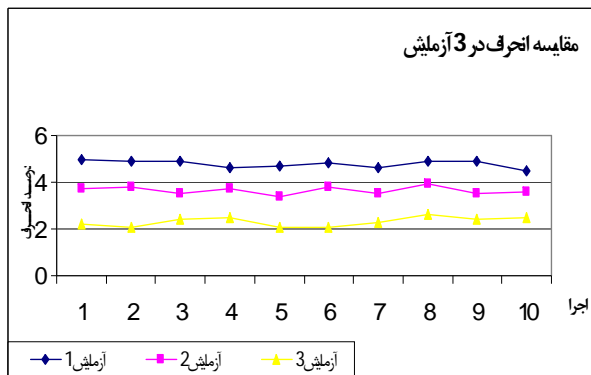
نام	OPT.	F.S.	B.S. (max)	B.S. Time (sec)	I.T. (sec)	S.B.	Deviation
آزمایش 1	مشاهده نشد	اولین تکرار	50 تکرار	15	510	ندارد	4.8
آزمایش 2	مشاهده نشد	اولین تکرار	50 تکرار	15	617	ندارد	3.6
آزمایش 3	مشاهده نشد	اولین تکرار	50 تکرار	15	1915	ندارد	2.3



شکل 4: انحراف از بهینه: مثال J309_1، آزمایش 2



شکل 5: انحراف از بهینه: مثال J309_1، آزمایش 3



شکل 6: انحراف از بهینه: مثال J309_1، در هر سه آزمایش

نتیجه گیری

در مورد مثال 301_1 بعد از حداکثر 12 زمانبندی در 10 بار اجرای الگوریتم به جواب بهینه دست یافته شد و برای مثال J309_1 بعد از حداکثر 50 زمانبندی تولید شده در 10 بار اجرای الگوریتم به بهترین جواب دست یافته شد و میانگین انحراف‌ها از بهینگی برای مثال J309_1 برابر 2.3 درصد به دست آمد. تعداد زمانبندی‌های تولید شده در هر بار اجرای الگوریتم 2000 عدد بود.

الگوریتم در مثال J301_1 که درجه سختی متوسطی داشت، در آزمایش‌های 2 و 3 به جواب بهینه

الگوریتم مورچگان ارایه شده بر مثال‌های J301_1 و J309_1 آزمایش شد و نتایج با بهترین جواب‌های یافته شده تا کنون مقایسه شدند. در کل تکرارهای اجرا شده روی دو مثال (20000 تکرار در مثال J301_1 و 20000 تکرار در مثال J309_1) حالت ایستا مشاهده نشد. اطلاعات ابتکاری نرمالیزه شده، قانون انتخاب احتمالی نوین، تبخیر فرمون کم، تعداد تکرارهای زیاد و تعریف مناسب همسایگی شدنی، همگی کارایی الگوریتم را در جهت تولید جواب‌های با کیفیت افزایش دادند.

نشان داده است که این مطلب حاکی از تنظیمات مناسب عوامل الگوریتم، پویا بودن عوامل با گذشت تکرارهای الگوریتم، استفاده از قانون ابتکاری نرمالیزه و دو نرخ بودن تبخیر فرمون است. در مجموع می‌توان گفت الگوریتم، هدف اصلی طراحی خود را که داشتن کارآیی لازم از نظر کاوش کردن فضای حل بدون حذف ناحیه‌ای و تولید جواب‌های با کیفیت بالا بوده است برآورده می‌کند.

دست پیدا کرد، اما در مورد مثال J309_1 که زمان‌برترین مثال در بین مسایل الگو بود، به جواب بهینه دست پیدا نکرد. البته این موضوع فرصت خوبی بود تا بتوان اثر تغییرات عوامل را بر رفتار الگوریتم و انحراف از بهینگی آن مطالعه کرد. به روشنی در شکل (6) مشاهده می‌شود که الگوریتم در آزمایش 3 (نهایی) انحراف‌های کمتری نسبت به بهینه

مراجع

- 1- Sprecher, A. (2002). "Network decomposition techniques for resource-constrained project scheduling." *Journal of the Operational Research Society*, Vol. 53, PP. 405-414.
- 2- Brucker, P., Drexel, A., M?hring, R. H., Neumann, K. and Pesch, E. (1999). "Resource-constraint project scheduling: Notation, classification, models, and methods." *European Journal of Operational Research*, Vol. 112, No. 1, PP. 3-41.
- 3- Yang, B., Geunes, J. and O'Brien, W. (2006). "Resource constrained project scheduling; past work and new directions." *Research Report*, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL.
- 4- Davis, E.W. and Patterson, J. H. (1975). "A comparison of heuristic and optimum solutions in resource-constrained project scheduling." *Management Science*, Vol. 21, No. 8, PP. 944-955.
- 5- Ulusoy, G. and ?zdamar, L. (1989). "Heuristic performance and network/resource characteristics in resource-constrained project scheduling." *Journal of Operational Research Society*, Vol. 40, PP. 1145-1152.
- 6- Herroelen, W., De Reyck, B. and Demeulemeester, E. (1998). "Resource-constrained project scheduling: A survey of recent developments." *Computers and Operations Research*, Vol. 13, No. 4, PP. 279-302.
- 7- Kolisch, R. and Hartmann, S. (1999). *Heuristic algorithms for solving resource constrained project scheduling problem: classification and computational analysis*, in J. Weglarz (editor): *Project scheduling: Recent models, algorithms and applications*, Kluwer, Amsterdam, the Netherlands, PP. 147-178.
- 8- Kolisch, R., Schwindt, C. and Sprecher, A. (1998). *Benchmark instances for project scheduling problems*, in J. Weglarz (ed.), *Handbook on recent advances in project scheduling*, Kluwer, PP. 197-212.
- 9- Dorigo, M. and Stutzle, T. (2004). *Ant colony optimization*, Bradford Book, MIT Press.
- 10- Besten, M. D., Stützle, T. and Dorigo, M. (2000). "Ant colony optimization for the total weighted tardiness problem." in *Lecture Notes in Computer Science, Berlin, Germany: Springer-Verlag*, Vol. 1917, PP. 611-620.
- 11- Bottee, H. M. and Bonabeau, E. (1998). "Evolving ant colony optimization." *Adv. Complex System*, Vol. 1, No. 2-3, PP. 149-159.
- 12- Dorigo, M. and Gambardella, L.M. (1997). "Ant colony system: A cooperative learning approach to the traveling salesman problem." *IEEE Trans. Evol. Comput.*, Vol. 1, PP. 53-66.
- 13- Valckenaers, P., Van Brussel, H., Kollingbaum, M. and Bochmann, O. (2001). "Multi-agent coordination and control using Stigmergy applied to manufacturing control." *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Vol. 2086, PP. 317-334.
- 14- Dorigo, M., Maniezzo, V. and Colorni, A. (1996). "The ant system: Optimization by a colony of cooperating agents." *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, Vol. 26, No. 1, PP. 29-4.

- 15- Bauer, A., Bullnheimer, B., Hartl, R.F. and Strauss, C. (1999). "An ant colony optimization approach for the single machine total tardiness problem." *In Proceeding of Congress on Evolutionary Computation*, PP. 1445-1450.
- 16- Merkle, D. and Middendorf, M. (2000). "An ant algorithm with a new pheromone evaluation rule for total tardiness problems." *Lecture Notes in Computer Science*, Berlin, Germany: Springer-Verlag, Vol. 1803, PP. 287-296.
- 17- Leu, S.-S. and Hung, T.-H. (2002). "A genetic algorithm-based optimal resource-constrained scheduling simulation model." *Construction Management and Economics*, Vol. 20, No. .
- 18- Kall, P. and Wallace, S. W. (1994). *Stochastic programming*, John Wiley and Sons.
- 19- Beck, J.C. and Wilson, N. (2005). "Proactive algorithms for scheduling with probabilistic durations." *The Proceedings of the Nineteenth International Joint Conferences on Artificial Intelligence*, Edinburgh, Scotland, 30 July-5 August, PP. 1201-1206.
- 20- Merkle, D., Middendorf, M. and Schmeck, H. (2002). "Ant colony optimization for resource-constrained project scheduling." *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 4, PP. 89-98.

واژه های انگلیسی به ترتیب استفاده در متن

- 1- Resource Constrained Project Scheduling Problem
 - 2- Non-Deterministic Polynomial Hard
 - 3- Benchmark Instance
 - 4- Chance Constrained Programming
 - 5- Joint Chance Constrained Programming
 - 6- Activity on Node (AON)
 - 7- Exploitation
-