فصلنامه علمی ° پژوهشی مطالعات مدیریت صنعتی ° سال دوازدهم، شماره ۳۵، زمستان ۹۳

صفحات ۸۲ - ۶۱

حل مسئله زمان‌بندی پروژه بر پایه حجم کار با منبع محدود برای کمینه کردن هزینه‌های زودکرد و دیرکرد با الگوریتم ژنتیک

امیر گلاب[۱]، امیرعباس نجفی[۲]

(دریافت:۹۲/۱۲/۳ پذیرش:۹۳/۳/۵)

چکیده:

در این مقاله، مسئله زمان‌بندی پروژه بر پایه حجم کار[۳] با محدودیت منبع برای کمینه کردن هزینه‌های زودکرد و دیرکرد با توجه به حجم کار ثابت برای فعالیت‌ها، روابط پیش‌نیازی و محدودیت منابع تجدیدپذیر مورد بررسی و تحلیل قرارگرفته است. باتوجه به اینکه فعالیت‌ها دارای سررسید مشخصی هستند برای فعالیت‌هایی که از سررسید خود انحراف دارند جریمه زودکرد و دیرکرد در نظر گرفته می شود. برای حل این مسئله دو الگوریتم فراابتکاری ژنتیک و شبیه‌سازی تبرید طراحی شده و پارامترهای این الگوریتم‌ها با استفاده از روش تاگوچی تنظیم شده است. برای بررسی عملکرد این دو الگوریتم، مجموعه‌ای از مسائل نمونه حل و با با استفاده از آزمون‌های آماری به مقایسه جواب‌های حاصل از دو الگوریتم پرداخته شده است.

واژگان کلیدی: زمان‌بندی پروژه، زودکرد و دیرکرد، الگوریتم ژنتیک، الگوریتم شبیه‌سازی تبرید، حجم کار

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی

۱- کارشناس ارشد مهندسی صنایع ، دانشگاه آزاد اسلامی،واحد علوم و تحقیقات، گروه مهندسی صنایع ، قزوین، ایران
.a.golab@qiau.ac.ir

۲- استادیار دانشکده مهندسی صنایع، دانشگاه صنعتی خواجه نصیرالدین طوسی ، تهران، ایران.
amirg_01@yahoo.com

3 -Work-content

مقدمه

مسئله مطرح شده در این تحقیق از جمله مسائل زمانبندی پروژه با محدودیت منبع است که پایان فعالیت‌ها دارای سررسید مشخصی هستند و انحراف از سررسید پرداخت جریمه به همراه خواهد داشت،در این گونه مسائل شبکه فعالیت‌ها از نوع گره ای تعریف می‌شود و سطح دسترسی به منابع، محدود است. موضوع بارز این پژوهش، طرح موضوع حجم کار فعالیت‌ها در مسئله زمانبندی پروژه با محدودیت منابع است؛ به عنوان مثال در پروژه‌های ساختمانی فعالیت‌های پروژه عموماً حجم کار ثابتی دارند که به صورت منبع-روز(مثلا ۲۰ نفر-روز) بیان می‌شوند، درنتیجه مدت زمان انجام فعالیت‌هایی که بر پایه حجم کار استوار هستند می‌تواند متغیر بوده و به میزان استفاده از منبع در روزهای انجام فعالیت بستگی دارد. مسئله زمانبندی پروژه با محدودیت منابع از نوع مسائل $NP-Hard$ در مسائل زمانبندی محسوب می‌شود .(بلاژویچ و لنسترا ،۱۹۸۳)

در گذشته تحقیقاتی بر روی این گونه مسائل صورت گرفته که با توجه به مسئله مورد بررسی در این مقاله به سه دسته قابل تقسیم است:

(الف) در زمینه زمان‌بندی پروژه با محدودیت منابع مطالعات متعددی انجام شده که برخی از مطالعات انجام شده بر روی مدل کلاسیک آن در ادامه آمده است. اگروال و همکاران (۲۰۱۱)یک دیدگاه ژنتیک عصبی که ترکیبی از الگوریتم ژنتیک و دیدگاه شبکه عصبی[1] بود، برای حل مسئله زمانبندی پروژه با منبع محدود ارائه کردند. اتلی و کهرامان (۲۰۱۲) یک حل جهت مسئله زمانبندی فعالیت‌ها با منبع محدود با استفاده از الگوریتم جستجوی ممنوع بیان کردند که هدف آنها کمینه کردن زمان پایان پروژه بود. مندز و همکاران (۲۰۰۹) الگوریتم ژنتیکی برای حل مسئله زمانبندی پروژه با منبع محدود ارائه کردند که کروموزوم این الگوریتم بر پایه کلید تصادفی[2] استوار بود. در الگوریتم آنها از یک قانون اولویت ابتکاری[3] نیز

1.Neural- network (NN)
2.Random keys
3.Heuristic Priority Rule

استفاده شده است. تیسنگ و چن (۲۰۰۶) یک الگوریتم ترکیبی فراابتکاری[1] شامل الگوریتم کلونی مورچگان، الگوریتم ژنتیک و استراتژی جستجوی محلی برای مسئله زمانبندی پروژه با محدودیت منبع ارائه کردند. جیرو و همکاران (۲۰۱۰) یک الگوریتم ژنتیک بر اساس کروموزوم چند آرایه برای مسئله زمانبندی فعالیت‌ها با منبع محدود ارائه کردند.

(ب) در خصوص مسئله زمانبندی پروژه با منابع محدود و در نظرگیری هزینه‌های زودکرد و دیرکرد مطالعات انجام شده عبارتند از: ونهوک و همکاران (۲۰۰۱) یک روش شاخه و حد[2]را برای زمانبندی فعالیت‌ها با منبع محدود و هزینه‌های زودکرد و دیر کرد بیان کردند. در این مقاله فعالیت‌ها دارای زمان پایان مشخص، جریمه زودکرد و دیرکرد و منبع تجدید پذیر محدود هستند. افشار نجفی و شادرخ (۲۰۰۸) یک زمانبندی پروژه برای کمینه کردن مجموع هزینه‌های زودکرد و دیرکرد فعالیت‌ها با در نظرگرفتن روابط پیش‌نیازی و زمان تکمیل مشخص برای فعالیت‌ها ارائه کردند.خوش‌جهان و همکاران(۲۰۱۳) مسئله زمانبندی پروژه را با هدف کمینه‌سازی هزینه‌های زودکرد و دیرکرد با محدودیت منابع مورد توجه قرار داده و برای حل مدل آن دو الگوریتم فرابتکاری ارائه دادند. رنجبر و همکاران (۲۰۱۲) یک روش حل بهینه بر مبنای الگوریتم شاخه و حد برای مسئله زمانبندی پروژه با منبع محدود با هدف کمینه کردن هزینه‌های دیرکرد ارائه کردند. شادرخ و کیانفر (۲۰۰۷) یک الگوریتم ژنتیک برای حل مسئله سرمایه‌گذاری در منابع[3] ارائه کردند که دیرکرد پروژه با جریمه همراه خواهد بود. شهسوار و همکاران (۲۰۱۰) مسئله سرمایه‌گذاری در منابع پروژه با جریان‌های نقدی تنزیل شده را در شرایطی که برای زمان اتمام پروژه سر رسید مشخص شده و ساختار پاداش- جریمه وجود دارد، مورد بررسی قرار دادند.

(ج) در ارتباط با زمانبندی پروژه با در نظرگیری مفهوم حجم کار، مطالعات محدودی انجام شده است. رنجبر و کیانفر (۲۰۱۰)یک الگوریتم ژنتیک برای مسئله زمانبندی فعالیت‌ها با محدودیت منبع با پروفایل کاری قابل تغییر با هدف کمینه‌سازی مدت زمان پروژه مطرح

1.Hybrid Metaheuristic
2.Branch and Bound
3.Resource Investment Problem

کردند. فاندلینگ و تراتمن (۲۰۰۷) از نوعی قانون اولویت برای زمانبندی فعالیت‌ها با حجم کار ثابت استفاده کردند، هدف آن‌ها کمینه سازی زمان اجرای پروژه بود و برای استفاده از منبع حجم کار برای هر فعالیت، حدود پایین و بالا در نظر گرفتند.

با توجه به مطالعات فوق می توان اعلام کرد که ترکیب مسئله زمانبندی پروژه بر پایه حجم کار و همچنین زمانبندی پروژه با هدف کمینه سازی هزینه‌های زودکرد- دیرکرد، تاکنون مورد مطالعه قرار نگرفته است. لذا در این تحقیق، مسئله زمانبندی پروژه با محدودیت منابع بر پایه حجم کار و با هدف کمینه سازی هزینه‌های زودکرد- دیرکرد مورد بررسی قرار گرفته و جهت حل آن دو الگوریتم فرا ابتکاری توسعه داده می شود.

مدل ریاضی مسئله

مسئله مورد بررسی در این تحقیق، مسئله زمانبندی پروژه با محدودیت منبع[1] است که در آن حجم کار فعالیت‌ها ثابت می باشد. در این مسئله یک منبع به عنوان منبع حجم کار در نظر گرفته شده و سایر منابع (در صورت وجود) تحت تاثیر منبع حجم کار می‌باشند. هر دو دسته، منبع محدود بوده و دارای سقف در دسترس، در طول اجرای پروژه هستند. هدف مدل نیز کمینه‌سازی کل هزینه زود کرد و دیر کرد فعالیت‌ها است. با توجه به توضیحات ارائه شده نمادهای مورد استفاده در این مدل به شرح جدول (۱) است:

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی

1.RCPSP: Resource-Constrained Project Scheduling Problem

جدول ۱. معرفی نمادهای مورد استفاده در مدل

نماد	شرح نماد	نماد	شرح نماد
i	اندیس فعالیت ها	t	اندیس زمان
n	تعداد فعالیت‌ها	r_{it}	میزان منبع مورد نیاز فعالیت i در زمان t
V	مجموعه فعالیت‌ها	w_i	حجم کار تعریف شده برای فعالیت i
T	مجموعه زمان ها	R	سقف منبع حجم کار
d_i	مدت زمان فعالیت i	R_k	سقف منبع وابسته به منبع حجم کار نوع k
h_i	سر رسید فعالیت i	LB_{r_i}	حدپایین استفاده ازمنبع حجم کار در طول انجام فعالیت i
f_i	زمان اتمام فعالیت i	UB_{r_i}	حدبالااستفاده ازمنبع حجم کار در طول انجام فعالیت i
s_i	زمان شروع فعالیت i	LB_{d_i}	حد پایین مدت فعالیت i
e_i	هزینه هر واحد زودکرد فعالیت i	UB_{d_i}	حد بالا مدت فعالیت i
t_i	هزینه هر واحد دیرکرد فعالیت i	lst_i	دیرترین زمان شروع فعالیت i
E_i	مدت زودکرد فعالیت i	est_i	زودترین زمان شروع فعالیت i
T_i	مدت دیرکرد فعالیت i	$P(j)$	مجموعه فعالیتهای پیش‌نیاز فعالیت j
$\mathcal{R}$	مجموعه منابع وابسته به منبع حجم کار	Z	مقدار تابع هدف

اگر h_i زمان سررسید فعالیت i و f_i زمان تکمیل فعالیت i باشد، بر این اساس، مدت زودکرد و دیرکرد یک فعالیت به ترتیب به صورت زیر خواهند بود:

$$E_i = \max(0, h_i - f_i) \qquad (i \in v) \qquad (1)$$

$$T_i = \max(0, f_i - h_i) \qquad (i \in v) \qquad (2)$$

اگر t_i, e_i به ترتیب هزینه زودکرد و دیرکرد فعالیت i باشد آنگاه مجموع هزینه‌های زودکرد و دیرکرد فعالیت i به شکل زیر بیان می‌شود:

$$e_i E_i + t_i T_i \qquad (i \in v) \qquad (3)$$

دو معادله (۴)و (۵) به ترتیب بیان کننده حد پایین و حد بالای مدت زمان انجام فعالیت i هستند:

$$LB_{d_i} = \lceil w_i / UB_{r_i} \rceil \qquad (i \in v) \qquad (4)$$

$$UB_{d_i} = \lceil w_i / LB_{r_i} \rceil \qquad (i \in v) \qquad (5)$$

محدودیت (۶)بیان می‌کند که مجموع منبع حجم کار مورد نیاز فعالیت i از شروع تا پایان برابر حجم کار فعالیت i است در این تحقیق $r_{it} \in Z_{\geq 0}$ و $w_i \in Z_{>0}$خواهند بود:

$$\sum_{t=S_i}^{f_i} r_{it} = w_i \qquad (i \in v) \qquad (6)$$

محدودیت (۷) بیان می‌کند منبع مورد نیاز فعالیت i قبل از شروع و بعد از پایان برابر صفر است:

$$\sum_{t=0}^{S_i - 1} r_{it} = \sum_{f_i+1}^{T} r_{it} = 0 \qquad (i \in v) \qquad (7)$$

محدودیت(۸) بیانگر محدودیت منبع حجم کاری می باشد که در طول مدت اجرای پروژه برقرار است.

$$\sum_{i \in V} r_{it} \leq R \qquad (t \in T) \qquad (8)$$

محدودیت (۹) بیانگر محدودیت مربوط به سقف در دسترس سایر منابع پروژه است. میزان استفاده فعالیت‌ها از این دسته از منابع، به میزان استفاده از منبع حجم کار وابسته می‌باشد لذا در معادله مربوطه مقدار استفاده هر فعالیت از این منابع به صورت تابعی از مقدار استفاده فعالیت از منبع حجم کاری (r_{it}) نوشته می شود.

$$\sum_{i \in V} r_{ik}(r_{it}) \leq R_k \qquad (t \in T) \qquad (9)$$

محدودیت (۱۰) تضمین می‌نماید فعالیت i در مدت زمان انجام، منقطع نشود.

$$LB_{r_i} \leq r_{it} \leq UB_{r_i} \qquad t = (s_i, \dots, f_i) \qquad (10)$$

نامساوی شماره (۱۱) بیانگر این است که مدت انجام فعالیت i مقداری مابین حد پایین و حد بالای زمان انجام فعالیت iباشد.

$$LB_{d_i} \leq d(i) \leq UB_{d_i} \qquad (i \in v) \qquad (11)$$

محدودیت شماره (۱۲)مشخص می کند که زمان شروع فعالیت i بین زودترین و دیرترین زمان شروع آن باشد. زودترین و دیرترین زمان شروع فعالیت‌ها از محاسبات متداول رفت و برگشت مربوط به روش مسیر بحرانی بدست می‌آید. لیکن برای محاسبه est_i در محاسبات رفت از LB_{d_i} و برای محاسبه lst_i در محاسبات برگشت از UB_{d_i} استفاده می‌شود.

محدودیت (۱۳) ارتباط بین زمان شروع و پایان فعالیت ها را نشان می دهد.

$$est_i \leq s_i \leq lst_i \qquad (i \in v) \qquad (12)$$

$$f_i = s_i + d_i \qquad (i \in v) \qquad (13)$$

محدودیت (۱۴)روابط پیش‌نیازی فعالیت‌ها بیان می‌شود. اگر فعالیت i پیشنیاز فعالیت jباشد آنگاه زمان شروع فعالیت jبزرگتر از زمان پایان فعالیت i خواهد بود :

$$S_j \geq f_i \qquad \forall_i \epsilon P(j)\,, \ (i \in v) \qquad (14)$$

تابع هدف مدل نیز کمینه‌سازی مجموع جریمه های زودکرد و دیرکرد فعالیتها می‌باشد.

$$Min\ Z = \sum_{i=1}^{n} (e_i E_i + t_i T_i) \qquad (15)$$

با توجه به آنکه مسئله مورد بررسی در زمره مسائلNP-Hard است، لذا در بخش‌های بعدی به معرفی و طراحی دو الگوریتم ژنتیک و شبیه سازی تبرید برای حل مدل مذکور پرداخته خواهد شد.

الگوریتم ژنتیک

الگوریتم ژنتیک مجموعه بسیار بزرگی از حل‌های ممکن تولید می‌کند هریک از حل‌ها بااستفاده ازتابع تناسب[1]، مورد ارزیابی قرار می‌گیرند آنگاه تعدادی از بهترین حل‌ها باعث تولید حل‌های جدیدی می‌شوند که این کار باعث تکامل حل‌ها می‌شود، در هر مرحله تعدادی از اعضای جمعیت قبل به همان صورت برای نسل بعدی مورد استفاده قرار می‌گیرند و برخی با استفاده از اپراتورهای ژنتیکی نظیر تقاطع[2] و جهش[3] برای تولید فرزندان به کار می‌روند. (گلدنبرگ ،۱۹۸۹)

شبه کد الگوریتم ژنتیک طراحی شده در شکل (۱) قابل مشاهده است:

Initial GA control parameter
While (termination condition not met)
While (termination criterion is not satisfied) do
Generate initial population
Generation succession activity list (a)
Generation work-content profile (p)
Generate member of population
Evaluate fitness initial population
End while
Select mate randomly
Apply two-point crossover operation to build $child1$ and $child2$
Apply mutation operation to build $child$ by $pattern$
Evaluate fitness
Update elite set ByElitism
$Z^* = Best\ solution$
End while

شکل۱. شبه کد الگوریتم ژنتیک

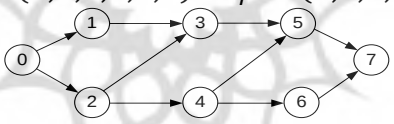
1 - Fitness
2 -Crossover
3 -Mutation

در ادامه به معرفی اجزاء الگوریتم ژنتیک طراحی شده پرداخته خواهد شد.

تعیین پروفایل حجم کار و لیست توالی فعالیت‌ها

جهت ایجاد کروموزوم، ابتدا دو جزء (۱) پروفایل حجم کار(۲) لیست توالی فعالیت‌ها تشکیل می‌شود. بخش اول مربوط به پروفایل حجم کار، یک ماتریس دارای n سطر و m ستون است که سطرها بیانگر فعالیتها و ستونها بیانگر زمان می‌باشند. لذا nبیانگر تعداد فعالیتها و m مشخص کننده حداکثر مقدار ممکن مدت زمان فعالیت‌های پروژه می باشد. درایه ij این ماتریس بیانگر مقدار حجم کاری تخصیص داده شده به فعالیت i در روز j ام اجرای آن می‌باشد. تولید پروفایل حجم کار به اینگونه است که می بایست مجموع سطر iام پروفایل برابر حجم کار مشخص شده فعالیت iام شودو هر درایه سطر iام پروفایل عضو $Z_{\geq 0}$ و بین حد بالا و حد پایین استفاده از منبع حجم کار آن باشد. به عنوان پروژه مثال نشان داده در شکل ۲ را در نظر بگیرید. لازم بذکر است که فعالیت های شماره صفر و هفت، فعالیت موهومی بوده و بیانگر شروع و پایان پروژه می باشند.

$$w = (6,8,12,4,3,12) UB_d = (3,4,4,2,3,4)$$
$$LB_r = (2,2,3,2,1,4) UB_r = (4,5,5,4,9,6)$$



شکل ۲. شبکه AONپروژه ای شامل ۶ فعالیت اصلی

یک پروفایل حجم کار شبکه شکل ۲، به صورت شکل ۳ خواهد بود. تعداد سطرهای این ماتریس برابر ۶ (تعداد فعالیت‌های پروژه) و تعداد ستونها برابر ۴ (حداکثر مدت زمان فعالیت‌های پروژه$Max\ UB_d$) می باشد.

3	2	1	0
3	4	1	0
3	3	3	3
4	0	0	0
1	2	0	0
6	6	0	0

شکل۳. یک پروفایل ممکن برای شبکه مثال

بخش دوم لیست توالی فعالیت‌ها است که بیانگر ترتیب فعالیت‌ها در تولید کروموزوم مربوطه است. این لیست می‌بایست بنحوی تهیه شود که پیش‌نیازهای هر فعالیت درایه‌های قبلی قرار گیرند. بعنوان نمونه، شکل ۴ نشان دهنده یک لیست توالی فعالیت برای مثال مورد بررسی است.

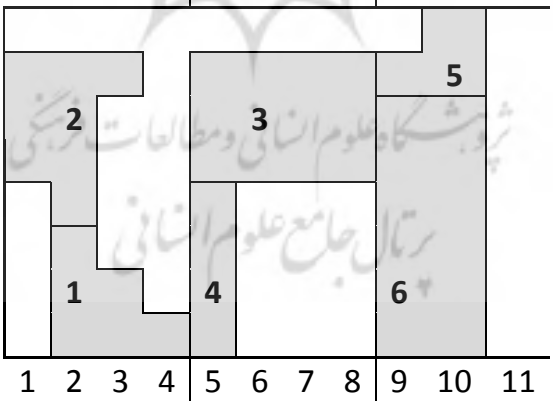
1	2	4	3	6	5

شکل۴. یک نمونه لیست توالی فعالیت

طرح کدگذاری و ارزیابی کروموزوم‌ها

با درنظرگیری دو بخش یاد شده (پروفایل حجم کار و لیست توالی فعالیت‌ها) هر کرموزوم از طریق زمانبندی فعالیت‌ها به روش تولید زمانبندی موازی[1] استخراج می‌گردد. باتوجه به بخش اول، کروموزوم به نحوی که محدودیت منابع رعایت شود تعیین شده و با استفاده از بخش دوم، پروفایل استفاده فعالیت از منابع استخراج می‌گردد. شکل ۵ نمونه‌ای از کروموزوم ایجاد شده برای مثال شکل ۲ است.

0	3	2	1	0	0	0	0	0	0
3	4	1	0	0	0	0	0	0	0
0	0	0	0	3	3	3	3	0	0
0	0	0	0	4	0	0	0	0	0
0	0	0	0	0	0	0	0	1	2
0	0	0	0	0	0	0	0	6	6



شکل۵. کروموزوم مثال

1 - Parallel Schedule Generation Scheme

سپس اعضا توسط تابع برازش ژنتیک که در این مسئله همان تابع هدف می‌باشد براساس زمان پایان و سررسید فعالیت‌ها مورد سنجش قرار می‌گیرد؛ به عبارت دیگر مجموع هزینه‌های زودکرد و دیرکرد فعالیت‌ها محاسبه خواهد شد. در این مرحله نقاط تقاطع کروموزوم‌ها نیز شناسایی می‌گردد.

عملگر تقاطع

انتخاب والدین برای تولید فرزندان به روش رولت ویل[1] انجام می‌پذیرد. در عملگر تقاطع، دو نقطه تقاطع از نقاط شناسایی شده از یک والد به صورت تصادفی انتخاب می‌گردد. سپس در والد دیگر همین دونقطه به صورت متناظر در نظر گرفته می‌شود؛ به عنوان مثال اگر دو نقطه تقاطع اول و دوم از والد اول به صورت تصادفی انتخاب شوند، دو نقطه اول و دوم از والد دوم نیز مد نظر خواهد بود. سپس عمل تقاطع دو نقطه‌ای بر روی دو والد صورت می‌پذیرد، بدین ترتیب به تعداد مورد نیاز (حاصل ضرب احتمال تقاطع در اندازه جمعیت) فرزند از عملگر تقاطع حاصل خواهد شد بلافاصله فرزندان حاصل از تقاطع ارزیابی می‌شوند.

با توجه به توضیحات ارائه شده نمونه عملگر تقاطع دو نقطه‌ای در شکل (۶) آورده شده است:

4	2	2	0	0	0	0	0	0	0	0
4	2	0	0	0	0	0	0	0	0	0
0	0	0	5	3	3	1	0	0	0	0
0	0	0	0	3	1	0	0	0	0	0
0	0	0	0	0	0	0	2	1	0	0
0	0	0	0	0	0	0	5	5	2	0

والد ۱

4	2	2	0	0	0	0	0	0	0	0
4	2	0	0	0	0	0	0	0	0	0
0	0	0	0	0	2	2	0	0	0	0
0	0	0	4	3	3	2	0	0	0	0
0	0	0	0	0	0	0	2	1	0	0
0	0	0	0	0	0	0	5	5	2	0

فرزند ۱

0	2	2	2	0	0	0	0	0	0	0
4	2	2	0	0	0	0	0	0	0	0
0	0	0	0	0	2	2	0	0	0	0
0	0	0	0	4	3	3	2	0	0	0
0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	5	6	1

والد ۲

0	2	2	2	0	0	0	0	0	0	0
4	2	2	0	0	0	0	0	0	0	0
0	0	0	0	5	3	3	1	0	0	0
0	0	0	0	0	3	1	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	5	6	1

فرزند ۲

شکل۶. نمونه عملگر تقاطع دو نقطه ای

1 - Roulette wheel

عملگر جهش

عملگر جهش اینگونه است که ناحیه‌ای که قرار است جهش روی آن صورت گیرد با انتخاب تصادفی یک نقطه از نقاط تقاطع شناسایی شده به غیر از نقطه پایانی آغاز می‌شود، ناحیه بین این نقطه تصادفی و نقطه بعدی، ناحیه جهش در نظر گرفته می‌شود. قسمت مشخص شده از این کروموزوم (والد) با ناحیه مشابه کروموزوم دیگر(الگو) که به صورت تصادفی از اعضای همان نسل انتخاب شده جایگزین می‌شود و بدین‌ترتیب به تعداد مورد نیاز (حاصل ضرب احتمال جهش در اندازه جمعیت) فرزند از عملگر جهش حاصل خواهد شد که فرزندان ایجاد شده بلافاصله مورد ارزیابی قرار می‌گیرند.

با توجه به توضیحات ارائه شده، نمونه عملگر جهش در شکل (۷) آورده شده است:

شکل۷. نمونه عملگر جهش

جمعیت نسل بعدی و شرط خاتمه

جهت تولید نسل بعدی، جمعیت نسل فعلی با فرزندان حاصل از عملگرهای تقاطع و جهش، ادغام می‌گردند. سپس به تعداد اندازه جمعیت از بهترین جواب‌ها جهت تشکیل نسل انتخاب می‌شوند. شرط خاتمه برای الگوریتم مذکور، تعداد نسل است؛ به طوری‌که الگوریتم پس از تولید تعداد نسل معینی پایان می‌یابد و جواب نهایی الگوریتم، برابر بهترین جواب نسل پایانی خواهد بود.

الگوریتم شبیه‌سازی تبرید

مکانیزم الگوریتم بدین گونه است که ابتدا از یک جواب اولیه آغاز می‌کند و سپس در حلقه تکرار و دمای ثابت جواب‌های همسایه تولید می‌شوند. اگر جواب همسایه بهتر از جواب فعلی باشد آن را به عنوان جواب فعلی قبول می‌کند، در غیر این صورت آن جواب مشروط پذیرفته خواهد شد. سپس دما به تدریج کاهش داده می‌شود. با کاهش تدریجی دما، در گام‌های پایانی، احتمال کمتری برای پذیرش جواب‌های بدتر وجود خواهد داشت. بدین ترتیب الگوریتم به سمت یک جواب بهتر حرکت می کند.

شبه کد الگوریتم شبیه سازی تبرید طراحی شده در شکل (۸) قابل مشاهده است:

Initial SA control parameter
While (termination criterion is not satisfied) do
Generate initial Answer Bank
Generation succession activity list (a)
Generation work-content profile (p)
Generate member of Answer Bank
Evaluate Answer Bank
End while
Set $Te = Te_0 = max(Z_1, Z_2, \ldots., Z_{nPop})$; set Z^* = select randomly from population
While (termination condition not met)
Generate solution Z in neighborhood of Z^* ; evaluate $Delta = f(Z) - f(Z^*)$
If $Delta \leq 0$; $Z^* = Z$
Else generate a random number $Ran \in (0,1)$
If $Ran \leq e^{(-\frac{Delta}{Te})}$; $Z^* = Z$
End; End
Recuse the Te
End while

شکل۸. شبه کد الگوریتم شبیه سازی تبرید

در ادامه به معرفی اجزاء الگوریتم شبیه سازی تبرید طراحی شده پرداخته خواهد شد.

جواب و دمای اولیه

در الگوریتم شبیه سازی، تبرید طراحی شده جهت تعیین جواب اولیه ابتدا همانند الگوریتم ژنتیک، مجموعه‌ای ازجواب‌ها (همانند کروموزم‌ها) تولید می‌َشود که این مجموعه را بانک جواب، نامگذاری می‌کنیم. تمام اعضای موجود مورد ارزیابی قرار می‌گیرند که در این ارزیابی، مجموع هزینه‌های زودکرد و دیرکرد و نقاط تقاطع هر عضو تعیین می‌گردد. سپس یکی از جواب‌ها به صورت تصادفی انتخاب شده، تا به عنوان جواب فعلی در نظر گرفته شود. لازم است دمای اولیه تا حد امکان بالا تعیین شود. دمای اولیه از رابطه زیر که در آن Te_0 دمای اولیه تعریف شده است تعیین می‌گردد.

$$Te_0 = max(Z_1, Z_2, \dots., Z_{nPop})$$

تولید جواب همسایه و نحوه پذیرش آن

روش تولید جواب همسایه، مشابه عملگر جهش در الگوریتم ژنتیک است؛ بدین ترتیب که ناحیه تغییر که مابین دو نقطه تقاطع متوالی در جواب فعلی است تعیین می‌شود، سپس یک عضو از بانک جواب‌ها به صورت تصادفی انتخاب شده تا ناحیه تغییر تعیین شده در جواب فعلی با ناحیه مشابه از جواب دیگر جایگزین شود. جواب همسایه مذکور بلافاصله مورد ارزیابی قرار می‌گیرد.

اختلاف مقدار جواب همسایه و فعلی Delta نامگذاری می‌شود اگر $Delta \leq 0$ باشد جواب همسایه بهتر است و جایگزین جواب فعلی می‌گردد اما اگر $Delta > 0$ باشد جواب همسایه به طور مشروط قابل پذیرش است از این رو مقدار تابع بولتزمان را با توجه به دمای فعلی Te محاسبه می‌کنیم:

$$P = e^{(-\frac{Delta}{T})}$$

سپس عددی تصادفی در بازه [0 1] تولید کرده و آن را Ran نامگذاری می‌کنیم. اگر $Ran \leq P$ بود، جواب همسایه جایگزین جواب فعلی خواهد شد.

تعداد تکرار و شرط خاتمه الگوریتم

در الگوریتم مذکور تعداد مشخصی تکرار در هر دما در نظر گرفته شده است که پس از تثبیت جواب می‌بایست دما کاهش یابد. از این رو دمای جدید توسط ضریب مشخصی حاصل شود:

$$Te_{new} = \alpha * Te_{old}$$

در صورتی‌که شرط خاتمه الگوریتم(تکرار الگوریتم شبیه سازی تبرید به تعداد مشخص) محقق شده باشد الگوریتم پایان می‌یابد و جواب نهایی نیز بهترین جواب محاسبه شده در تکرار پایانی خواهد بود.

تنظیم پارامترها و مقایسه الگوریتم‌ها

تنظیم پارامترها

تنظیم و تخصیص پارامترهای مناسب در کیفیت جواب الگوریتم‌ها تاثیر بسزایی دارد؛ لذا از روش تاگوچی جهت این امر استفاده می‌شود. تاگوچی توانست یک طراحی آزمایش‌ها را به شکلی ایجاد کند که تعداد آزمایش‌ها در مسئله کاهش داشته باشد. بر اساس این روش زمانیکه مسئله از نوع بهینه سازی و پاسخ تا حد ممکن کوچک مورد نظر است از رابطه زیر استفاده خواهیم کرد که در آن منظور از S مقادیر مطلوب و N مقادیر غیر مطلوب در نظر گرفته شده است .

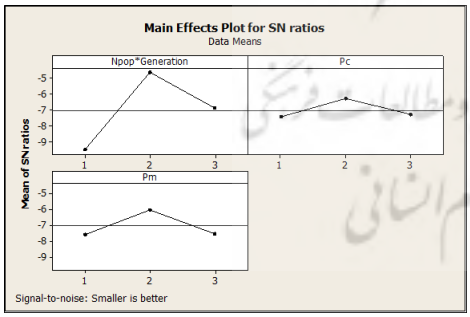
$$SN=-10\log(1/n \sum_{i=1}^{n} y_i^2)$$

آزمایشات جهت تنظیم پارامترهای الگوریتم ژنتیک و شبیه سازی تبرید در سه سطح انجام شد.

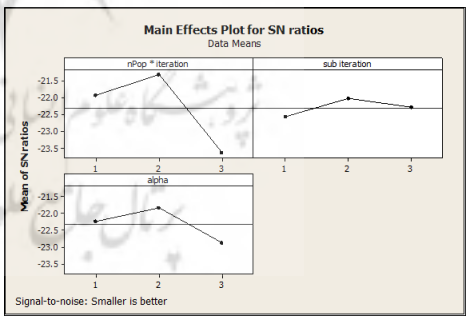
جدول ۲. پارامترهای مورد آزمایش دو الگوریتم توسط طراحی آزمایشات تاگوچی

پارامترهای مورد آزمایش برای الگوریتم ژنتیک			
احتمال جهش	احتمال تقاطع	(اندازه جمعیت ، تعداد نسل)	سطح
0.1	0.6	(75,50)	1
0.15	0.65	(50,75)	2
0.2	0.7	(38,100)	3
پارامترهای مورد آزمایش برای الگوریتم شبیه سازی تبرید			
ضریب کاهش دما	تکرار در دما ثابت	(اندازه بانک جواب، تعداد تکرار الگوریتم)	سطح
0.85	10	(100,20)	1
0.9	15	(125,16)	2
0.95	20	(150,13)	3

طراحی آزمایشات به روش تاگوچی در نرم‌افزار Minitab16 به شکل آرایه‌های متعامد $L9$ مورد آزمون قرار گرفت که نتیجه در شکل‌های ۹ و ۱۰ گزارش شده است.



شکل۹.نمودار S/N الگوریتم ژنتیک



شکل۱۰.نمودار S/N شبیه سازی تبرید

همانطور که مشاهده شد پارامترهای مناسب برای دو الگوریتمGASA به قرار زیر خواهد بود:

جدول ۳ .پارامترهای تایید شده برای دو الگوریتم جهت حل مسائل نمونه

پارامترهای مورد تایید جهت الگوریتم ژنتیک			
احتمال جهش	احتمال تقاطع	(اندازه جمعیت ، تعداد نسل)	سطح
0.15	0.65	(50,75)	2
پارامترهای مورد تایید جهت الگوریتم شبیه سازی تبرید			
ضریب کاهش دما	تکرار در دما ثابت	(اندازه بانک جواب، تعداد تکرار الگوریتم)	سطح
0.9	15	(125,16)	2

حل مسائل نمونه

نتایج حاصل از آزمایش الگوریتم ژنتیک و شبیه سازی تبرید ارائه شده در قالب حل ۳۰ مسئله برای هر الگوریتم در جدول ۴ ارائه شده است، حل مسئله نرم افزار متلب برنامه نویسی شده است .

جدول ۴. میانگین حل ۳۰ مسئله با دو الگوریتم ژنتیک و شبیه سازی تبرید (هر دسته دارای ۵ مسئله)

زمان پردازش الگوریتم شبیه سازی تبرید	زمان پردازش الگوریتم ژنتیک	مقدار تابع هدف الگوریتم شبیه سازی تبرید	مقدار تابع هدف الگوریتم ژنتیک	
3.70	4.48	3	0	مسائل دارای ۵ فعالیت(میانگین حل ۵ مسئله)
4.86	5.67	9.8	1.4	مسائل دارای ۱۰ فعالیت (میانگین حل ۵ مسئله)
5.36	6.65	9.4	5	مسائل دارای ۱۵ فعالیت (میانگین حل ۵ مسئله)

مسائل دارای ۲۰ فعالیت (میانگین حل ۵ مسئله)	9.6	21.2	5.81	4.96
مسائل دارای ۳۰ فعالیت (میانگین حل ۵ مسئله)	10.6	19.2	7.49	5.73
مسائل دارای ۴۰ فعالیت (میانگین حل ۵ مسئله)	30.8	38.2	9.17	6.72
میانگین کل	9.6	16.8	6.54	5.22

قابل مشاهده است که میانگین جواب‌های الگوریتم ژنتیک بهتر است درحالی که میانگین زمان حل شبیه سازی تبرید کمتر است. در ادامه از آزمون t زوجی جهت مقاسیه دو الگوریتم بهره خواهیم برد اما قبل از آن می‌بایست از نرمال بودن جواب‌های هر دسته اطمینان حاصل کنیم .

تست نرمال بودن جواب‌ها

جهت انجام آزمون نرمال بودن از آزمون‌های شاپیرو ° ویلک[1] و کولموگروف ° اسمیرنوف[2] در سطح خطای ۰/۰۵ از نرم افزار ***SPSS*** استفاده شده است.

جدول ۵. نتایج تست شاپیرو ° ویلک و کولموگروف ° اسمیرنوف

شاپیرو ° ویلک	کولموگروف ° اسمیرنوف	
سطح معناداری	سطح معنا داری	الگوریتم
0.09417	0.10710	ژنتیک
0.52	0.2	شبیه سازی تبرید

از آنجا که سطح معناداری آزمون‌ها بیشتر از سطح معنا داری ۰/۰۵ است، لذا با اطمینان ۰/۹۵ فرض نرمان بودن جواب‌ها تایید می‌گردد.

1 -Shapiro-Wilk

2 -Kolmogorov-Smirnov

آزمون t زوجی

با توجه به نرمال بودن داده‌ها از آزمون t زوجی جهت تست معناداری اختلاف میانگین‌ها استفاده خواهد شد . به بیان آماری:

$H_0: \mu 1 = \mu 2$: اختلاف میانگین‌ها معنادار نیست

$H_1: \mu 1 \neq \mu 2$: اختلاف میانگین‌ها معنادار است

جدول ۶. نتایج آزمون t زوجی

آزمون t زوجی			
	Pair	**t**	**Sig**
Grand total	GA-SA	-8.8101	1.076536396162188E-9

همانطور که در جدول ۵ مشاهده می‌شود، سطح معناداری آزمون(sig)برای دو مورد آزمایش تقریبا صفر و کمتر از ۰/۰۵ است؛ لذا با اطمینان ۰/۹۵ فرض H_0رد می‌شود و کیفیت جواب حاصل از الگوریتم ژنتیک بهتر از جواب‌های حاصل از الگوریتم شبیه‌سازی تبرید است.

حل مدل به روش دقیق

جهت اطمینان از عملکرد الگوریتم ژنتیک مذکور، ۵ مسئله که دارای پنج فعالیت هستند توسط نرم افزار *Lingo* حل شد:

جدول ۷. مقایسه خروجی‌های نرم افزار Lingo و الگوریتم ژنتیک

مقدار تابع هدف لینگو	مقدار تابع هدف الگوریتم ژنتیک	
0	0	مسئله ۱
0	0	مسئله ۲
0	0	مسئله ۳
0	0	مسئله ۴
0	0	مسئله ۵

همانطور که در جدول ۷ قابل مشاهده است هر دو دارای جواب مساوی و برابر صفر هستند. از این رو می‌توان از عملکرد مناسب الگوریتم اطمینان حاصل نمود.

نتیجه گیری

در این تحقیق مسئله زمانبندی پروژه بر پایه حجم کار با منبع محدود برای کمینه کردن هزینه‌های زودکرد و دیرکرد بررسی شد. با توجه به $NP-hard$ بودن مسئله از دو الگوریتم ژنتیک و شبیه‌سازی تبرید برای حل مدل بهره گرفته شد، جهت تنظیم پارامترهای دو الگوریتم از روش تاگوچی استفاده شد و به منظور مقایسه کارایی دو الگوریتم ۳۰ مسئله در ۶ دسته ارائه شد تا کیفیت جواب‌ها با استفاده از آزمون t زوجی مورد مقایسه قرار گیرد که کیفیت جواب‌های حاصل از الگوریتم ژنتیک بهتر از الگوریتم شبیه سازی تبرید بود، درحالی‌که زمان حل الگوریتم شبیه سازی تبرید تا حدی از زمان حل الگوریتم ژنتیک کمتر است. برای تحقیقات آتی حل مدل با دو تابع هدف، به عبارت دیگر اضافه کردن ارزش خالص فعلی هزینه‌ها پیشنهاد می‌گردد.

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی

منابع:

AfsharNajafi, B., Shadrokh, Sh., An Algorithm for the weighted Earliness-Tardiness unconstrained project scheduling program,*Journal of Applied Sciences*,2008, 1651-1659

Agarwal,A., Colak,S., Erenguc S., A Neurogenetic approach for the resource-constrained project scheduling problem , *Computers & Operations Research* 38 (2011) 44–50

Atli, O. , Kahraman, C., Fuzzy resource-constrained project scheduling using taboo search algorithm, *International Journal of Intelligent Systems* , Volume 27, Issue 10, October 2012, Pages 873-907

Blazewicz J, Lenstra JK, RinnooyKan AHG. Scheduling projects to resource constraints: classification and complexity. *Discrete Applied Mathematics* 1983; 5:11–24.

Fündeling, N. Trautmann; A priority-rule method for project scheduling with work-content constraints, *European Journal of Operational Research*, 2010; 568-574

Goldeberg, D.E, "Genetic Algorithm in Search" *Optimization and Machine Learning Wesley*, Reading, MA, 1989.

Jairo R. Montoya-Torres, Edgar Gutierrez-Franco, Carolina Pirachica´n-Mayorga, Project scheduling with limited resources using a genetic algorithm, *International Journal of Project Management* 28 (2010) 619–628

Khoshjahan, Y, Najafi, AA, Afshar-Nadjafi, B,Resource constrained project scheduling problem with discounted earliness–tardiness penalties: Mathematical modeling and solving procedure , *Computers & Industrial Engineering*, Volume 66, Issue 2, October 2013, Pages 293-30086–101

Lin-Yu Tseng, Shih-Chieh Chen,. A hybrid metaheuristic for the resource-constrained project scheduling problem, *European Journal of Operational Research* 175 (2006) 707–721

Mendes, J.F. Gonçalves, M.G.C. Resende, A random key based genetic algorithm for the resource constrained project scheduling problem, *Computers & Operations Research* 36 (2009) 92 – 109

Ranjbar,M, Khalilzadeh, M, Kianfar, F, Etminani, K , An optimal procedure for minimizing total weighted resource tardiness penalty costs in the resource-constrained project scheduling problem , *Computers & Industrial Engineering*, Volume 62, Issue 1, February 2012, Pages 264-270

Ranjbar, M., Kianfar, F. Resource-constrained project scheduling problem with flexible work profiles: A genetic algorithm approach, *ScientiaIranica* , Volume 17, Issue 1 E, June 2010, Pages 25-35

Shadrokh,Shahram,Kianfar,Fereydoon. A genetic algorithm for resource investment project scheduling problem, tardiness permitted with penalty.*European Journal of Operational Research* 181 (2007)

Shahsavar M, Niaki STA, Najafi AA, An efficient genetic algorithm to maximize net present value of project payments under inflation and bonus-penalty policy in resource investment problem. *Advances in Engineering Software*, 41, 1023-1030, 2010.

Vanhoucke, M., Demeulemeester E., Herroelen, W., an Exact Procedure for the Resource-Constrained Weighted Earliness–Tardiness Project Scheduling Problem, *Annals of Operations Research*, 2001,102, 179–196