

زمانبندی ماشین‌های موازی برای کارهای قابل تقسیم با در نظر گرفتن زمان راه‌اندازی وابسته به توالی

فریبرز جولای

دانشیار گروه مهندسی صنایع - پردیس دانشکده های فنی - دانشگاه تهران

رضا توکلی مقدم

دانشیار گروه مهندسی صنایع - پردیس دانشکده های فنی - دانشگاه تهران

فرشید آزادیان

کارشناس ارشد مهندسی مدیریت نساجی دانشگاه صنعتی امیرکبیر

(تاریخ دریافت ۸۴/۵/۲۴، تاریخ دریافت روایت اصلاح شده ۸۴/۱۲/۱۳، تاریخ تصویب ۸۵/۱/۲۶)

چکیده

در این مقاله به مساله زمانبندی کارهای قابل تقسیم بر روی ماشین‌های موازی یکسان با هدف کمینه‌سازی دیرکرد کل پرداخته می‌شود. در این مساله هر یک از ماشین‌ها برای قبول کار جدید نیاز به راه‌اندازی دارد که زمان آن وابسته به توالی کارهاست. در اینجا با بررسی روش‌های مختلف و ارائه و اثبات قضایایی در خصوص نحوه تقسیم کار و زمانبندی آنها، یک الگوریتم ابتکاری معرفی می‌شود و سپس کارایی آن با نتایج مدل ریاضی مساله برای چندین مساله نمونه با استفاده از داده‌های تصادفی مقایسه می‌گردد.

واژه‌های کلیدی: زمانبندی ماشین‌های موازی یکسان، دیرکرد کل، زمان راه‌اندازی، کارهای قابل تقسیم، الگوریتم ابتکاری

مقدمه

[۷]، نشان داد که در حالت ماشین‌های موازی مساله دست‌کم از نوع غیر چند جمله‌ای دودویی است. در مورد معیار کمینه‌سازی دیرکرد وزنی لاولر و همکاران [۹] ثابت نمودند زمانی که کارها دارای اوزان مختلف باشند مساله غیر چند جمله‌ای سخت است.

یکی از رایج‌ترین روش‌های قطعی در حل این دسته از مسایل، رهیافت برنامه‌ریزی پویا می‌باشد [۸،۱۰]، هر چند که وسعت کاربرد این روش به دلیل تنوع و تعدد گزینه‌ها چه در زمان تخصیص کارها و چه هنگام تعیین توالی آنها بر روی هر ماشین، بسیار محدود است. رهیافت انشعاب و تحدید نیز در حل این مساله بکار گرفته شده است که از آن جمله می‌توان به تحقیقات المغربی و پارک [۱۱] برای کارهایی با موعد تحویل و زمان پردازش یکسان، بارن و برنن [۱۲] برای حداکثر ۲۰ کار و ۴ ماشین و نیز یالوی و چوو [۱۳] اشاره نمود. گذشته از این راهکارها، تحقیقاتی نیز برای حل این دسته از مسایل در شرایط خاص و یا در زمینه‌های مشابه صورت گرفته است؛ نظیر تحقیق لاولر

ماشین‌های موازی به عنوان یکی از زیرمجموعه‌های اصلی و پایه در فن زمانبندی از جایگاه ویژه و مهمی برخوردارند و همواره زمانبندی این مدل بر مبنای معیارهای عملکرد مختلف مورد نظر محققین بوده است [۱-۳] با این توجه که، عموم روش‌های حل در مدل‌های پیچیده‌تر مانند مدل گردش کارگاهی بر مبنای راهکارهای مدل‌های ساده‌تر از جمله ماشین‌های موازی استوار است [۵،۴].

از آنجاکه همواره نمی‌توان راهکار پاسخیابی مطلق را در مدل ماشین‌های موازی یافت و برای بسیاری از معیارهای عملکرد، بویژه معیارهای مبتنی بر دیرکرد که با زمان تکمیل رابطه خطی ندارند، مساله از نوع غیرچندجمله‌ای (NP) می‌باشد. لنتسرا و همکاران [۶]، بر مبنای مساله تقسیم ثابت نمودند که مساله زمانبندی ماشین‌های موازی با هدف کمینه نمودن دیرکرد کل حتی برای دو ماشین نیز یک غیر چند جمله‌ای دودویی می‌باشد. کولاماس [۸] بر پایه تحقیق دوو و لنگ در مدل تک ماشین

پردازش یک جزء آغاز می‌شود دیگر نمی‌توان آن را متوقف ساخت به عبارت دیگر حق قطع کار شغله‌ای^۱ مجاز شمرده نمی‌شود. کارها بر اساس ویژگی‌هایشان به دسته‌هایی تقسیم شده‌اند که بطور قراردادی از این پس خانواده کار شناخته می‌شود؛ برای پردازش هر کار لازم است تا ماشین برای کار جدید آماده‌سازی شود و زمان این عمل وابسته به توالی کارها می‌باشد، اما در هر صورت این زمان برای کارهایی که در یک خانواده جای می‌گیرند نسبت به حالات دیگر کمتر خواهد بود.

در این مساله هر کار J از میان N کار تعیین شده، به P_j واحد زمان پردازش بر روی یک یا چند ماشین از میان m پردازنده موجود نیاز دارد. هر کار دارای موعد تحویل d_j می‌باشد و می‌توان آن را به دلخواه تقسیم و هر جزء را بطور مستقل پردازش نمود. برای پردازش کارهای مشابه نیاز به زمان راه اندازی برابر S_{sim} و برای غیرمشابه‌ها S_{nsim} واحد زمان عملیات راه‌اندازی و آماده‌سازی نیاز است $(S_{sim} < S_{nsim})$.

این مساله که حالت خاصی از مدل کلاسیک ماشین‌های موازی می‌باشد، بر گرفته از یک نمونه کاربردی در صنایع نساجی (خط بافندگی تار و پودی) است. در این مدل علاوه بر زمان راه‌اندازی وابسته به توالی، بحث تقسیم و تفکیک کارها و نحوه اجرا و برنامه‌ریزی آن نیز مطرح است که خود بر پیچیدگی مساله خواهد افزود. اگرچه پیچیدگی این مورد همچنان باز است، اما بنظر می‌رسد که مساله دست کم از نوع NP-hard می‌باشد.

بطور کلی در حل مسایل ماشین‌های موازی باید دو پرسش را پاسخگو بود: اولاً کارها به چه ترتیب و توسط چه ماشین‌هایی باید پردازش شوند و ثانياً توالی پردازش آنها بر روی ماشین‌های تخصیص یافته به چه صورت خواهد بود؛ بدیهی است نحوه انجام هر یک از این دو مساله و نقش متقابل آنها بر یکدیگر تاثیر مستقیم بر عملکرد برنامه نهایی خواهد داشت. حل همزمان این دو مورد از دلایل پیچیدگی مساله اصلی می‌باشد. در ادامه سعی می‌شود تا با استفاده از مجموعه‌ای از قضایا تا حد امکان این دو مساله را بطور مجزا و با مراعات اصل بهینگی، پاسخ‌یابی نمود. بدین منظور در ادامه ابتدا بحث تخصیص منابع و سپس تعیین توالی کارها بررسی می‌گردد و راهکار مناسبی برای حل مساله ارائه می‌شود.

[۱۴] با در نظر گرفتن مدل حمل و نقل و کارهایی با زمان پردازش برابر، روت [۱۵] برای کارهایی با موعد تحویل یکسان، و یا تحقیقات پریترسکر و همکاران [۱۶] که از روش برنامه‌ریزی خطی صفر - یک استفاده نموده‌اند.

نظر به پیچیدگی مساله زمانبندی ماشین‌های موازی و بزرگی ابعاد مسایل دنیای واقعی، روش‌های قطعی نمی‌توانند چندان راهگشا باشند و عموماً از راهکارهای ابتکاری در این موارد استفاده می‌شود. این روش‌ها اغلب بر تعیین توالی کارها بر اساس یک معیار تقدمی و سپس تخصیص کارها به ماشین‌ها بر مبنای زمان در دسترس بودن آنها، استوار است. روش‌های مختلفی در این زمینه ایجاد و ارائه شده است؛ از آنجمله می‌توان به MRM^1 توسط مونتر [۱۷]، $Covert^2$ توسط کارول [۲۰]، AU^3 توسط مورتن و همکاران [۱۹]، WI توسط ویلکرسون و ایروین [۲۰]، TCR^4 توسط هوو و چانگ [۲۱]، DS توسط دوگراماسی و سورکیس [۲۲]، KPM توسط کولاماس [۲۵]، $PDEC$ توسط کولاماس [۱۳]، HSA^5 توسط کولاماس [۲۳]، $HPJF^6$ توسط چن [۲۴]، MDD^7 اولین بار توسط بکر و برتراند [۲۵] سپس علی‌دای و روسا [۲۶] اشاره نمود.

در ادامه این مقاله، ابتدا مساله تعریف شده و سپس یک راهکار قطعی برای پاسخ‌یابی بهینه مساله در قالب مدل برنامه‌ریزی ریاضی ارائه می‌شود. سپس دو گام اصلی حل مساله یعنی تخصیص منابع و تعیین توالی، بررسی و قضایای مربوطه بیان و اثبات می‌گردد. در ادامه کارایی راهکار پیشنهادی توسط آزمون داده‌های تصادفی ارزیابی و با پاسخ‌های مدل ریاضی اعتبار سنجی می‌گردد.

تعریف مساله و ارائه مدل ریاضی آن

مساله مورد بررسی در این مقاله، زمانبندی کارهای قابل تقسیم بر روی مجموعه معینی از ماشین‌های موازی یکسان با در نظر گرفتن زمان راه‌اندازی وابسته به توالی، بر مبنای کمینه‌سازی معیار عملکرد دیرکرد کل می‌باشد. ماشین‌های موازی یکسان و همواره در دسترس بوده و در هر لحظه تنها قادر به پردازش یک کار می‌باشند. کارها را می‌توان به صورت بسته‌هایی با اندازه دلخواه تقسیم نمود و بطور مستقل و حتی بطور غیر همزمان مورد پردازش قرار داد، اما در هر صورت زمان تکمیل یک کار به عنوان زمان تکمیل آخرین جزء آن در نظر گرفته می‌شود. هنگامیکه

ماشینهایی که به پردازش آن کار اختصاص یافته‌اند منظور گردد. محدودیت (۴) الزامی می‌سازد که یک کار تنها یک بار بر روی یک ماشین انجام شود (به قضیه ۱ در بخش آتی رجوع شود). قید (۵) بیانگر آن است که هر کار باید کار مقدم و موخر داشته باشد. زمان تکمیل هر کار از محدودیت (۶) محاسبه می‌شود و دیرکرد به عنوان فزونی مثبت زمان تکمیل به موعد تحویل منظور می‌شود (۷). محدودیت های (۸) تا (۱۱) نیز متغیرهای مساله را تعریف می نمایند که شرح آن در صدر مدل درج شده است. در این مدل مقدار CUB را میتوان از دو برابر جمع کلیه زمانهای انجام کار بدست آورد.

به منظور پاسخ‌یابی مساله در حالت بهینه، در این بخش یک مدل ریاضی طراحی و ارائه می‌گردد. البته در این مدل مواردی فراتر از مساله مورد بحث در نظر گرفته شده است و مدل اخیر قابلیت استفاده در محیط ماشین‌های موازی مستقل و نامشابه را نیز دارد و بعلاوه بر خلاف زمان راه‌اندازی وابسته به توالی که در مساله مورد بحث منحصر به دو مقدار S_{sim} و S_{nsim} می‌گردد در این مدل امکان معرفی زمانهای راه‌اندازی مستقل برای هر ترکیبی از کارها در نظر گرفته شده است.

تابع هدف مدل (۱) به شکل کمینه سازی مقدار دیرکرد است. محدودیت (۲) متضمن انجام کل کار است. محدودیت (۳) اطمینان می‌دهد که اجزا هر کار تنها برای

$$\text{Min } \sum_{i=1}^N Z(i) \tag{1}$$

s.t.

$$\sum_{k=1}^M q(i, k) = Q(i) \quad i = 1 \dots N \tag{2}$$

$$q(i, k) \leq \left[\sum_{\substack{j=0 \\ j \neq i}}^N x(j, i, k) \right] \times CUB \quad i = 1 \dots N ; k = 1 \dots M \tag{3}$$

$$\sum_{\substack{i=0 \\ i \neq j}}^N x(i, j, k) \leq 1 \quad j = 0 \dots N ; k = 1 \dots M \tag{4}$$

$$\sum_{\substack{i=0 \\ i \neq h}}^N x(i, h, k) - \sum_{\substack{i=0 \\ i \neq h}}^N x(h, i, k) = 0 \quad h = 1 \dots N ; k = 1 \dots M \tag{5}$$

$$[x(i, j, k) - 1] \cdot CUB + C(i, k) + S(i, j) + q(j, k) \times p(j) \leq C(j, k) \quad \begin{matrix} i = 0 \dots N \\ j = 1 \dots N ; i \\ k = 1 \dots M \end{matrix} \tag{6}$$

$$C(i, k) - d(i) \leq Z(i) \quad i = 1 \dots N ; k = 1 \dots M \tag{7}$$

$$x(i, i, k) = 0 \quad i = 1 \dots N ; k = 1 \dots M \tag{8}$$

$$C(0, k) = 0 \quad k = 1 \dots M \tag{9}$$

$$x(i, j, k) \in \{0, 1\} \quad i, j = 0 \dots N ; k = 1 \dots M \tag{10}$$

$$q(i, k) \geq 0 \quad C(i, k) \geq 0 \quad Z(i) \geq 0 \quad i = 1 \dots N ; k = 1 \dots M \tag{11}$$

که با افزایش تعداد تقسیمات نسبت زمان‌های غیر تولیدی افزایش می‌یابد؛ بدین ترتیب لازم است تا حالت بهینه‌ای را تعیین نمود.

قضیه ۱. اگر دو کار i و j با توالی از پیش تعیین شده $i \rightarrow j$ بر روی یک پردازنده در نظر گرفته شوند، توالی که در آن کلیه اجزا کار i پیش از زواقع شود برای هر معیار عملکرد

تخصیص منابع به کارها

یکی از گام‌های اصلی در حل این مساله تعیین نحوه تخصیص منابع موجود بین کارها می‌باشد؛ در این ارتباط تاثیر تقسیم کار به اجزای کوچکتر نیز شایان توجه است، چرا که با تقسیم بیشتر کارها انعطاف در برنامه‌ریزی افزایش می‌یابد اما از سوی دیگر بحث زمان راه‌اندازی مطرح است

عادی بهینه خواهد بود.

اثبات. در کارهای قابل تقسیم، زمان تکمیل برابر زمان پایان پردازش آخرین جزء می‌باشد. بنابراین قرار گرفتن حتی یک جزء غیر تهی از کار Z در میان اجزا کار i موجب افزایش زمان تکمیل کار i خواهد شد، بعلاوه تقلیلی در زمان پردازش کار Z نیز رخ نمی‌دهد □

در قضیه فوق نحوه عمل برای یک پردازنده معین گردید، در قضیه زیر روش مواجهه با چندین ماشین موازی بررسی می‌گردد.

قضیه ۲. اگر کار Z بر روی m ماشین موازی یکسان پردازش شود، تخصیص بهینه شامل حالتی است که در آن زمان پردازش کار بر روی کلیه ماشین‌های اختصاص یافته بدون توجه به میزان و زمان آغاز پردازش، برابر باشد.

اثبات. بر مبنای قضیه فوق در تخصیص بهینه می‌توان میزان کار اختصاص یافته به هر ماشین را تعیین نمود. این مقادیر را به x_1, x_2, \dots, x_m نمایش داده می‌شوند. از آنجا که ممکن است ماشین‌ها همزمان پردازش کار را آغاز نکرده باشند، پس می‌توان انتظار داشت که این مقادیر با یکدیگر برابر نباشند؛ ولی زمان پایان کار کلیه ماشین‌ها برابر است و مجموع این اجزا برابر کل اندازه کار می‌باشد؛ زمان تکمیل کار i در این روش با C_i نمایش داده می‌شود.

حال اگر فرض شود که استفاده از راهکار قضیه فوق بهینه نمی‌باشد، آنگاه راهکار دیگری وجود دارد که مقادیر حاصل از آن بر روی دست‌کم یک ماشین متفاوت از مقادیر ارائه شده می‌باشد. حالات زیر را می‌توان متصور شد:

- دست‌کم بر روی یکی از ماشین‌ها مقدار کاری بیش از راهکار قضیه تخصیص داده شود. اگر این مقدار اضافه با Y نمایش داده‌شود. در این حالت اگر چه از میزان بار کاری یک یا چند ماشین دیگر کاسته می‌شود، اما از سوی دیگر زمان تکمیل کار i به صورت $C_i + Y$ افزایش می‌یابد.
- دست‌کم بر روی یکی از ماشین‌ها مقدار کاری کمتر از راهکار قضیه تخصیص داده شود. اگر این مقدار کاستی با Y نمایش داده شود؛ در این حالت لازم است تا کسری بوجود آمده توسط سایر ماشین‌ها جبران گردد و این بدان معناست که باید از ماشین جدیدی اضافه بر m ماشین استفاده نمود که ناقص فرض اولیه است و یا حداقل به میزان $Y/(m-1)$ به بار کاری ماشین‌ها و زمان تکمیل اضافه شود. بدین ترتیب این حالت نیز در

مجموعه غالب جای نخواهد گرفت □

در قضیه فوق اشاره شد که در حالت بهینه کلیه ماشین‌ها در یک لحظه مشخص از انجام کار فارغ می‌شوند، تعیین این لحظه به معیار ارزیابی عملکرد انتخابی وابسته است. به عنوان نمونه اگر هدف کمینه‌سازی حداکثر زمان تکمیل کارها در نظر گرفته شود، کران آرمانی برای این چنین مساله‌ای به صورت مجموع کل زمانهای تکمیل تقسیم بر تعداد پردازنده‌ها خواهد بود و بدین ترتیب برای حالت بهینه حد پایان کارها باید این مقدار منظور گردد. اما تعیین چنین کرانی برای مساله کمینه‌سازی دیرکرد کل دشوار است. نکته‌ای که می‌توان در مساله دیرکرد مورد توجه قرار داد، عامل موعد تحویل است. بدیهی است که پایان کار پس از موعد تحویل به منزله دیرکرد است، اما از سویی تکمیل کار پیش از این زمان نیز علاوه بر اینکه عموماً پاداشی در بر ندارد، موجب زیان (هزینه انبارداری، خواب سرمایه و غیره) خواهد بود. پس انتخاب موعد تحویل به عنوان کران پایان کارها بسیار مناسب بنظر می‌رسد.

انتخاب ماشین‌های پردازنده

نه تنها نحوه تخصیص منابع در عملکرد برنامه موثر است، بلکه انتخاب ماشینی که قرار است از آن در پردازش کارها استفاده شود با توجه به کار قبلی روی آن ماشین و زمان در دسترس بودنش بسیار مهم است. بسته به نوع کار پیشین، ماشین‌ها به دو دسته مشابه و غیر مشابه بر اساس کار جدید تقسیم می‌شوند، سپس در هر دسته بر مبنای زمان در دسترس بودن مرتب می‌گردند. منظور از زمان در دسترس بودن لحظه‌ای است که پس از راه‌اندازی‌های لازم می‌توان پردازش کار را آغاز نمود. به جهت پرهیز از انجام کار اضافه و کاستن از زمان‌های غیر تولیدی ابتدا از ماشین‌های مشابه استفاده می‌شود. ممکن است ماشین‌های مشابه برای انجام کار کافی نباشند، بدین صورت که با توجه به کران تعیین شده، یا با استفاده از کلیه ماشین‌ها نمی‌توان کار را به انجام رساند و یا تعدادی از آنها در زمانی پس از کران در دسترس قرار می‌گیرند؛ در چنین شرایطی می‌توان از ماشین‌های غیرمشابه نیز استفاده نمود، با این توجه که با استفاده از هر ماشین جدید مجموع کل ماشین‌های انتخابی (شامل مشابه و غیرمشابه انتخاب شده) تنها بر اساس زمان در دسترس بودنشان مرتب و مورد استفاده قرار می‌گیرند. در این حالت ممکن است با بکارگیری هر ماشین

آنگاه لازم است تا کران انتخابی تغییر یابد. در نهایت با استفاده از این الگوریتم: زمان تکمیل کار، ماشین‌های بکار گرفته شده و میزان و نحوه تخصیص کار به آنها معین می‌گردد.

تعیین توالی کارها

گام اساسی دوم در حل مساله تعیین توالی کارها، برای پردازش بر روی ماشین‌های موازی است. گذشته از پیچیدگی بحث تخصیص منابع در مورد مساله انتخابی، همانگونه که اشاره گردید، مساله برنامه‌ریزی کارهای مستقل و غیرقابل تقسیم بر روی ماشین‌های موازی حتی بدون زمان راه‌اندازی نیز یک مساله غیرچند جمله‌ای است. بر مبنای تحقیقات صورت گرفته بر عملکرد مقایسه‌ای این راهکارها، مشخص شده است که روشهای MDD و AU در مسایل ماشین‌های موازی از عملکرد نسبی بهتری برخوردارند. در این مقاله برای تعیین توالی، از راهکار ویژه‌ای متناسب با شرایط مساله مورد بحث استفاده شده است که بر مبنای اصول حاکم بر این روشها استوار است. روش "موعدهای تحویل اصلاح شده" توسط بکر و برتراند در سال ۱۹۸۲ [۲۵] ارائه شده و سپس در ۱۹۹۷ توسط علی‌دای و روسا [۲۶] اصلاح گردید. اساس کار این رهیافت مبتنی بر بازنویسی معادله هدف می‌باشد. هدف در مساله کمینه‌سازی دیرکرد کل را می‌توان چنین تعریف نمود:

$$\min Z = \sum_{j=1}^N \max \{C_j - d_j, 0\} \quad (12)$$

این معادله را می‌توان با اضافه و کسر نمودن یک مقدار معین چنین بازنویسی نمود:

$$Z' = Z \pm \sum_{j=1}^N d_j \quad (13)$$

$$Z' = \sum_{j=1}^N \max \{C_j, d_j\} - \sum_{j=1}^N d_j \quad (14)$$

از آنجا که موعد تحویل کارها مقادیر ثابتی هستند، می‌توان از بخش آخر تعریف Z' نیز صرف‌نظر نمود.

بنا به این روش، الگوریتم تخصیص کارها به ماشین‌ها، در این مقاله به صورت زیر بیان می‌شود:

۱- هر یک از کارها بر مبنای روش تخصیص منابع که پیشتر ذکر شد، مستقلاً به ماشین‌ها تخصیص می‌یابد و زمان تکمیل (C_j) هر یک محاسبه می‌گردد.

غیرمشابه جدید نیاز به استفاده از یک یا چند ماشین مشابه (که پس از آن در دسترس قرار می‌گیرند) از میان برود؛ به این نکته در الگوریتم تخصیص ماشین‌ها توجه شده است.

گاهی ممکن است با بکارگیری کلیه منابع موجود قابل استفاده، همچنان نتوان به کران تعیین شده پایبند بود، در این حالت دیرکرد غیر قابل اجتناب است و لازم است تا کران کمی افزایش یابد، اما در هر صورت باید کمترین کران ممکن مورد استفاده قرار گیرد که با استفاده از روشهای تکرار یا نصف کردن می‌توان به مقدار مناسب دست یافت.

الگوریتم تخصیص منابع

بر مبنای مطالب فوق‌الذکر الگوریتم تخصیص منابع برای یک کار در قالب زیر بیان می‌گردد:

۱- عامل موعد تحویل (d_j) به عنوان کران در نظر گرفته می‌شود.

۲- ماشین‌ها بر اساس کار پیشین به دو دسته مشابه و متفاوت با کار جدید تقسیم می‌شوند. ماشینی که قبل از این به پردازش هیچ کاری اشتغال نداشته است در دسته ماشین‌های غیر مشابه با کار جدید قرار می‌گیرد. دسته ماشین‌های مشابه انتخاب می‌شود.

۳- پردازنده‌های انتخاب شده بر مبنای زمان در دسترس بودن از زودترین به دیرترین مرتب می‌شوند. منظور از زمان در دسترس بودن لحظه‌ای است که آماده‌سازی ماشین با توجه به توالی کارهای مربوط به آن پایان یافته باشد. با شروع از ماشینی که در زودترین زمان در دسترس قرار می‌گیرد تا رسیدن به کران تعیین شده کار اختصاص داده می‌شود.

۴- مرحله پیش تا زمانی که کل کار تخصیص داده شود یا تمامی ماشین‌های مشابه بکار گرفته شوند و یا کار پایان نیافته ولی امکان بکارگیری ماشین دیگری از دسته مشابه (با توجه به زمان در دسترس قرار گرفتن ماشین نباشد) تکرار می‌گردد.

۵- اگر توقف در مرحله قبل به دلیل تخصیص کلیه ماشین‌های مشابه باشد، آنگاه با اضافه نمودن یک ماشین از دسته غیر مشابه به گروه پردازنده‌های انتخابی، برنامه‌ریزی از مرحله سوم مجدداً آغاز می‌شود.

این مرحله در صورت لزوم، تا زمانی که کلیه ماشین‌ها استفاده شوند و یا کار پایان یابد، تکرار می‌گردد.

۶- چنانچه توقف در مرحله قبل با پایان کار همراه نباشد،

۳ را پیشنهاد نموده‌اند. در جهت کاربردی ساختن این رابطه برای مسایل شامل زمان راه‌اندازی، معیار اولیه به صورت رابطه (۲۳) تکمیل گردید.

$$\pi_i = \left(\frac{w_i}{P_i} \right) \times EXP \left(-\frac{\max(0, d_i + C_i)}{k_i \bar{P}} \right) \times EXP \left(-\frac{S(i, j)}{k_2 \bar{S}} \right) \quad (23)$$

در این رابطه $S(i, j)$ زمان راه اندازی لازم برای پردازش کار j پس از i است و S میانگین زمان‌های راه‌اندازی است. دو عامل k_1 و k_2 متغیرهای تجربی می‌باشند که البته روابطی برای تخمین آنها بر مبنای داده‌ها ارائه شده است. برای بکارگیری دو الگوریتم MDD و AU به این صورت عمل می‌شود که در ابتدا فرض می‌شود هیچ کاری برنامه‌ریزی نشده است و سپس زمان تکمیل برای کلیه کارهای برنامه‌ریزی نشده طبق روش تخصیص منابع پیشنهادی محاسبه می‌شود. با جایگذاری مقادیر در روابط مربوط به هر کدام از دو روش که در همین بخش ارائه شده و با توجه به نحوه عمل هر یک، کاری را که در مقایسه برای پردازش بر دیگران برتری دارد انتخاب نموده و آنرا طبق روش مشخص شده تخصیص منابع برنامه‌ریزی می‌کنیم. این عملیات تا برنامه‌ریزی کلیه کارهای ادامه می‌یابد.

ارزیابی کارایی الگوریتم پیشنهادی در مسایل نمونه

برای مقایسه نسبی عملکرد دو روش MDD و AU آزمونی بر اساس داده‌های تصادفی و نزدیک به مسایل دنیای واقعی ترتیب داده شد. آزمون با استفاده از داده‌های تصادفی بر روی رایانه پنتیوم-۴ با نرم‌افزار به زبان دلفی بر روی سیستم عامل ویندوز صورت گرفت. در این آزمون کارها با موعد تحویل D با توزیع یکنواخت در بازه $[1, 2 \sum P_i / (qM)]$ در نظر گرفته شدند. مقدار q که معرف میزان فشردگی موعد تحویل کارها است با توزیع یکنواخت در بازه $[2, 5]$ تعیین شد. همچنین زمانهای راه اندازی نیز به صورت $S_{sim} \in U[3, 7]$ و $S_{nsim} \in U[12, 15]$ انتخاب شدند و کلیه کارها در دو دسته (خانواده) تقسیم گشتند. اما در خصوص زمان انجام کارها مشاهده شد که بین میانگین بازه زمانی انجام کارها و کارایی الگوریتم‌های MDD و AU رابطه‌ای وجود دارد. بدین منظور طی آزمایش جداگانه‌ای این رابطه مورد بررسی قرار گرفت. نتایج این آزمون در جدول (۱) و نمودارهای (۱) و (۲) مشاهده می‌گردد.

۲- بر مبنای زمان تکمیل، کارها به دو مجموعه u_1 و u_2 به صورت زیر تقسیم می‌گردند:

$$u_1 = \{i \in u : C_i > d_i\} \quad (15)$$

$$u_2 = \{i \in u : C_i \leq d_i\} \quad (16)$$

۳- دو مجموعه λ و γ طبق ضابطه زیر تشکیل می‌شود:

$$\lambda = \{i \in u_2 \mid d_i = \min(d_k) \forall k \in u_2\} \quad (17)$$

$$\gamma = \{i \in u_1 \mid C_i = \min(C_k) \forall k \in u_1\} \quad (18)$$

۴- دو کار a و b انتخاب می‌شوند به نحویکه:

$$C_a = \min\{C_k\} \mid k \in \gamma \quad (19)$$

$$d_b = \min\{d_k\} \mid k \in \lambda \quad (20)$$

۵- برای هر کار $(j \in a, b)$ مقدار δ_j تعیین می‌شود:

$$\delta_j = \min\{C_a, d_b\} \quad (21)$$

۶- کاری با کمترین مقدار δ_j به منابع موجود تخصیص می‌یابد.

در این روش دو مجموعه u_1 و u_2 به ترتیب گروه کارهای دیرکردار و بدون دیرکرد می‌باشند و دو مجموعه λ و γ ممکن است بیش از یک عضو داشته باشند. در واقع اساس این روش بر مبنای یک الگوریتم حریصانه است که در هر مرحله بهترین کار بعدی را بنحوی انتخاب می‌نماید که موجب کمترین افزایش در معیار دیرکرد (Z) گردد.

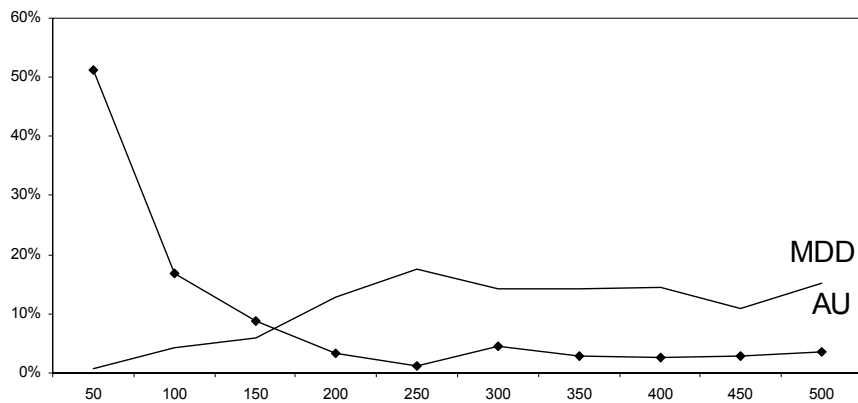
روش AU ابتدا توسط مورتون و همکاران [۱۴] برای مساله تک ماشین و سپس بوسیله لی و پیندو [۲۲] در محیط ماشینهای موازی و برای کارهای دارای زمان راه اندازی وابسته به توالی بکارگرفته شد. در این روش اولین ماشین در دسترس به کاری که دارای بیشترین مقدار معیار زیر باشد تخصیص می‌یابد:

$$\pi_i = \left(\frac{w_i}{P_i} \right) EXP \left(\frac{\max(0, d_i - t - P_i)}{k \bar{P}} \right) \quad (22)$$

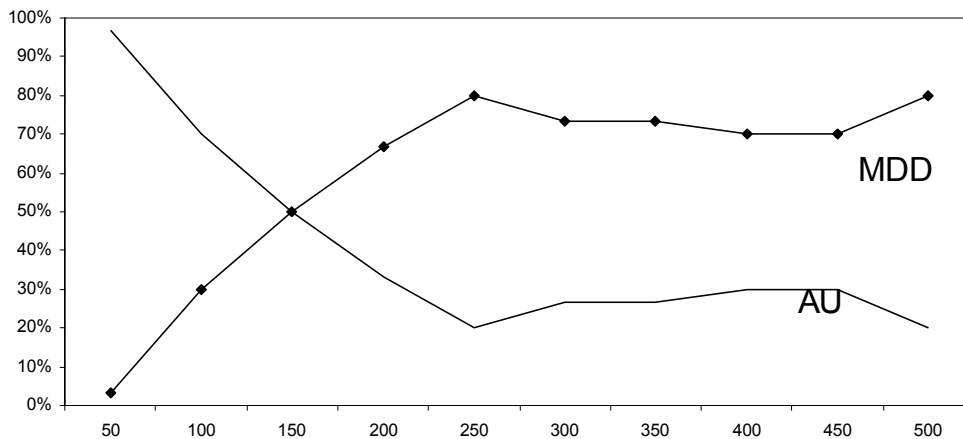
در این رابطه \bar{P} میانگین زمان پردازش کارها، k پارامتر پیش‌بینی و t زمان در دسترس بودن ماشین است. k میتواند مقادیر ۱ تا ۶ را بپذیرد ولی مبدعین این روش مقدار تجربی

جدول ۱: تاثیر زمان انجام کارها بر عملکرد الگوریتم ها.

زمان انجام کار	MDD				AU			
	PNOB%	Ratio%	MAR%	MIR%	PNOB%	Ratio%	MAR%	MIR%
50	96.7	0.7	19.9	0.0	3.3	51.2	103.8	0.0
100	70.0	4.3	45.1	0.0	30.0	16.8	64.3	0.0
150	50.0	6.0	78.5	0.0	50.0	8.7	41.3	0.0
200	33.3	12.7	69.5	0.0	66.7	3.3	20.2	0.0
250	20.0	17.7	120.4	0.0	80.0	1.2	14.7	0.0
300	26.7	14.3	48.0	0.0	73.3	4.6	58.9	0.0
350	26.7	14.2	67.0	0.0	73.3	2.9	27.2	0.0
400	30.0	14.5	85.3	0.0	70.0	2.5	22.3	0.0
450	30.0	10.9	42.0	0.0	70.0	2.8	32.7	0.0
500	20.0	15.2	51.3	0.0	80.0	3.5	75.2	0.0



نمودار ۱: میزان درصد نزدیکی به بهترین پاسخ نسبی بر مبنای کران زمان انجام کارها.



نمودار ۲: میزان درصد دستیابی به بهترین پاسخ نسبی بر مبنای کران زمان انجام کارها.

جدول ۲: عملکرد الگوریتمها برای ترکیبهای مختلف ماشینها و کارها بر اساس معیار دیرکردکل.

پردازنده	کار	MDD				AU			
		PNOB%	Ratio%	MAR%	MIR%	PNOB%	Ratio%	MAR%	MIR%
2	15	70.0	2.4	29.0	0.0	30.0	19.3	110.7	0.0
	20	80.0	2.1	23.2	0.0	0.0	10.9	34.4	0.0
	40	86.7	1.4	21.1	0.0	13.3	20.0	107.0	0.0
3	15	70.0	3.5	35.3	0.0	33.3	7.6	35.2	0.0
	20	50.0	4.6	25.4	0.0	50.0	8.6	43.3	0.0
	40	46.7	8.6	45.0	0.0	53.3	10.6	85.2	0.0

جدول ۳: عملکرد الگوریتم‌ها برای ترکیب‌های مختلف ماشین‌ها و کارها بر اساس معیار زمان تکمیل کل.

پردازنده	کار	MDD				AU			
		PNOB%	Ratio%	MAR%	MIR%	PNOB%	Ratio%	MAR%	MIR%
2	15	20.0	4.9	22.6	0.0	80.0	0.1	1.0	0.0
	20	13.3	4.6	20.3	0.0	86.7	0.1	1.8	0.0
	40	6.7	3.9	14.7	0.0	93.3	0.1	2.1	0.0
3	15	6.7	6.6	22.2	0.0	93.3	0.0	0.5	0.0
	20	3.3	5.6	15.0	0.0	96.7	0.0	0.2	0.0
	40	13.3	9.2	62.6	0.0	86.7	0.2	2.3	0.0

تعیین اثر ویژگی‌های کارها بر عملکرد الگوریتم‌ها آزمونهای جداگانه ای نیز بر روی مقدار q و نسبت بین زمان راه‌اندازی بیشینه به کمینه صورت گرفت؛ در آزمونهای اخیر که با استفاده از ۵ ماشین و برای ۲۰ کار در ۳۰ مساله مستقل انجام شد.

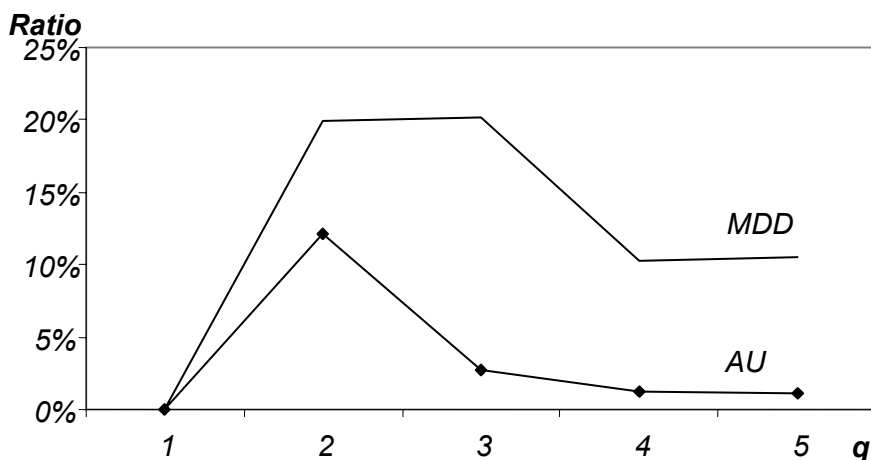
نتایج حاصل از آزمونهای فوق الذکر در جداول (۲) الی (۴) و نمودار (۳) ارائه شده است. همانگونه که در جدول (۲) مشاهده می‌گردد، بطور کلی نمی‌توان رابطه مشخصی بین افزایش تعداد کارها و کارایی الگوریتم‌ها (Ratio%) برقرار ساخت، هرچند که در برخی از موارد رابطه مستقیم افزایشی مشاهده می‌شود.

با افزایش تعداد ماشین‌ها عملکرد الگوریتم MDD اندکی تضعیف می‌شود در حالیکه کارایی روش AU، با وجود ضعیفتر بودن نسبت به راهکار دیگر، بهبود می‌یابد. در مجموع عملکرد نسبی روش MDD به مراتب برتر از رهیافت مبتنی بر AU است و بعلاوه پراکندگی پاسخ‌ها نیز کمتر است.

در این بررسی که بر روی ۵ پردازنده و با ۲۰ کار صورت گرفت، داده‌ها به روش فوق الذکر بدست آمدند و بازه زمان انجام کارها به عنوان متغیر به صورت پله‌ای از [20,50] تا [20,500] با دقت ۵۰ تغییر نمود. جهت تحلیل بهتر نتایج عملکرد هر الگوریتم بر مبنای درصد دفعات دستیابی به پاسخ بهینه (PNOB%)، نسبت بین پاسخ حاصله با بهترین پاسخ نسبی (Ratio%)، میزان بیشینه (MIR) و کمینه (MIR) نسبت بین پاسخ حاصله با بهترین پاسخ نسبی سنجیده شد.

بر اساس نتایج بدست آمده مشاهده می‌گردد که با افزایش کران بالای زمان انجام کارها به مقداری بیش از ۲۵۰ کارایی الگوریتم‌ها از ثبات نسبی برخوردار می‌شوند. پس نهایتاً برای اطمینان بیشتر بازه زمان انجام کارها برابر [20,370] انتخاب گردید.

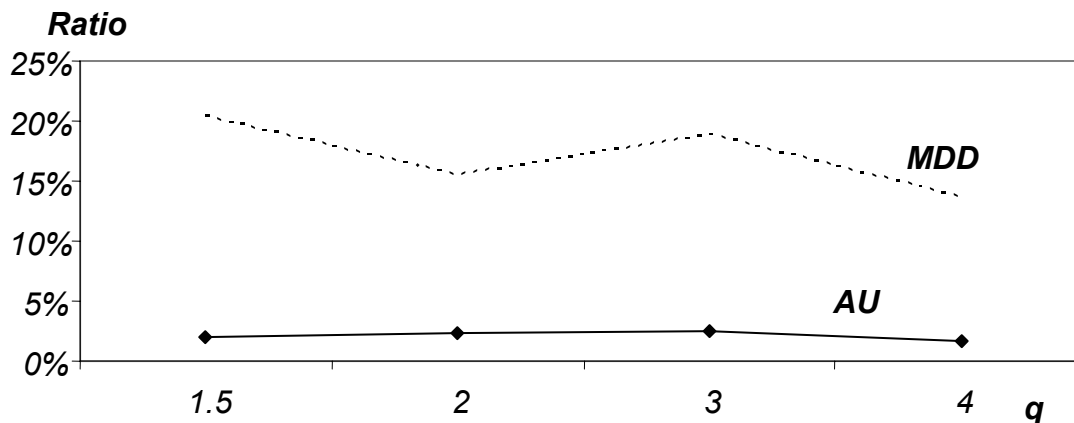
برای انجام آزمون از ترکیب‌های مختلف ۲، ۵ و ۱۰ پردازنده با ۱۵، ۲۰ و ۴۰ کار استفاده شد و برای هر حالت ممکن ۳۰ آزمون مستقل انجام شده و مقایسه برای الگوریتم‌های MDD و AU صورت گرفت. بعلاوه برای



نمودار ۳: اثر فشردگی موعد تحویل‌ها بر کارایی الگوریتم‌ها.

جدول ۴: اثر فشردگی موعد تحویلها بر کارایی الگوریتمها.

q	MDD				AU			
	PNOB%	Ratio%	MAR%	MIR%	PNOB%	Ratio%	MAR%	MIR%
1	86.7	-	-	-	16.7	-	-	-
2	46.7	19.9	157.3	0.0	0.0	12.1	146.8	0.0
3	26.7	20.2	77.0	0.0	73.3	2.7	25.0	0.0
4	20.0	10.3	30.0	0.0	80.0	1.3	11.4	0.0
5	16.7	10.6	35.4	0.0	83.3	1.1	10.2	0.0



نمودار ۴: اثر زمانهای راه اندازی بر کارایی الگوریتمها.

جدول ۵: اثر زمانهای راه اندازی بر کارایی الگوریتمها.

sim _i / nsim	MDD				AU			
	PNOB%	Ratio%	MAR%	MIR%	PNOB%	Ratio%	MAR%	MIR%
1.5	23.3	20.4	114.7	0.0	76.7	2.0	31.0	0.0
2	30.0	15.5	87.4	0.0	0.0	2.4	18.5	0.0
3	33.3	18.8	191.0	0.0	66.7	2.5	31.4	0.0
4	26.7	13.6	105.2	0.0	73.3	1.6	22.1	0.0

شدن موعد تحویل کارایی روش MDD بهبود یافته و سپس به حالت ثبات می‌رسد و مشابه این وضعیت برای راهکار AU نیز صادق است.

اما در مجموع برای q های بزرگتر و کارهایی با موعد تحویل بسته‌تر عملکرد روش AU بهتر است.

اثر زمانهای راه اندازی: در این آزمون به منظور ارزیابی اثر زمانهای راه اندازی بر الگوریتمهای پیشنهادی نسبت بین مقادیر زمان راه اندازی کارهای متفاوت به مشابه برای مقادیر ۱/۵، ۲ و ۳ متغیر فرض شد. با توجه به نتایج این آزمون که در جدول (۵) و نمودار (۴) آمده است ملاحظه می‌گردد که ارتباط مشخصی بین نسبت زمانهای راه اندازی و کارایی الگوریتمها وجود ندارد و عملکرد الگوریتمها تقریباً مستقل از زمان راه اندازی ها می باشد. لازم به ذکر است که

همچنین با وجود آنکه هدف این مقاله برنامه‌ریزی بر مبنای معیار دیرکرد است، اما آزمون مستقلی نیز برای بررسی کارایی این دو راهکار بر اساس معیار زمان تکمیل کل صورت گرفت که نتایج آن در جدول (۳) مشاهده می‌گردد. با توجه به این جدول مشخص می‌شود که اگر چه کارایی روش MDD قابل قبول بنظر می‌رسد اما راهکار AU عملکرد به مراتب بهتری دارد.

اثر موعد تحویل: در این قسمت عامل q برای مقادیر ۱ تا ۵ با فواصل واحد، متغیر در نظر گرفته می‌شود با این توضیح که برای مقادیر کوچکتر از آنجاکه ممکن است مساله بدون دیرکرد باشد، با توجه به رابطه مربوط به Ratio% امکان محاسبه این مقدار وجود ندارد. بر مبنای نتایج (جدول ۴ و نمودار ۳) مشاهده می‌گردد که با تنگ‌تر

گشتند. همچنین بازه زمان انجام کارها برابر [20,370] تعیین گردید.

برای انجام آزمون از ترکیب‌های مختلف ۲ و ۳ پردازنده با ۳ و ۴ و ۵ کار استفاده شد و برای هر حالت ممکن ۳۰ آزمون مستقل صورت گرفت و مقایسه برای الگوریتم‌های MDD و AU صورت گرفت. البته استثنا در مورد سه پردازنده و ۵ کار به علت پیچیدگی مساله دستیابی به پاسخ بهینه قطعی با استفاده از مدل ریاضی در زمان معقولی میسر نشد و از این مساله صرف‌نظر شد.

برای حل مدل ریاضی نیز از نسخه صنعتی برنامه لینگو (نسخه ۶) استفاده شد. نتایج حاصل از آزمون در جدول (۶) و (۷) ارائه شده است. در جدول ۶ برای هر یک از دو روش MDD و AU مقدار میانگین، حداکثر، حداقل و همچنین پراکندگی درصد اختلاف با پاسخ بهینه قطعی ارائه شده است. بعلاوه، تعداد تکرارهای لازم برای دستیابی به پاسخ بهینه مدل ریاضی نیز در جدول (۷) مشاهده می‌گردد.

همانگونه که در جدول (۶) مشاهده می‌گردد بطور نسبی کلیه معیارها با افزایش تعداد کار به آرامی رو به فزونی می‌روند و البته افزایش تعداد ماشین‌ها این روند را تسریع می‌کند. در مقام مقایسه در کلیه شاخص‌ها MDD عملکرد بهتری از خود نشان می‌دهد.

برای انتخاب نسبت پس از آزمون‌های مستقل جداگانه و اثبات عدم وابستگی نتایج، از نسبت‌های ۴، ۵، ۱۰، ۱۵ و ۲۰ به ۵ استفاده شد.

اعتبارسنجی الگوریتم پیشنهادی در مقایسه با پاسخ‌های مدل ریاضی

بطور کلی برای اعتبار سنجی الگوریتم‌ها سه روش وجود دارد: مقایسه با دنیای واقعی، مقایسه با بهترین پاسخ ممکن و مقایسه با روشهای توسعه یافته پیشین. از آنجاکه تحقیق مشابهی تاکنون صورت نگرفته است روش سوم کارایی ندارد. بعلاوه کران پاسخ بهینه نیز معمولا از ساده‌سازی مدل ریاضی مساله بدست می‌آید که در این مورد خاص ساده‌سازی لاگرانژین مدل نیز خود بسیار پیچیده است. بنابراین در نهایت ارزیابی الگوریتم با مقایسه نتایج حاصل از آن با پاسخ بهینه قطعی صورت پذیرفت.

با توجه به مدل ریاضی مورد استفاده ملاحظه می‌شود که به دلیل پیچیدگی مساله حل مدل تنها با تعداد محدود کارها و ماشین‌ها میسر است. داده‌های این آزمون به طور تصادفی مطابق روش اشاره شده در بند ۶-۱ تهیه شدند. در این آزمون مقدار q که معرف میزان فشردگی موعد تحویل کارها است با توزیع یکنواخت در بازه [2,5] و زمان‌های راه اندازی نیز به صورت $S_{sim} \in U[3,7]$ و $S_{nsim} \in U[12,15]$ انتخاب شدند. کلیه کارها در دو دسته (خانواده) تقسیم

جدول ۶: مقایسه کارایی الگوریتم‌ها با پاسخ بهینه قطعی حاصل از مدل ریاضی.

پردازنده	کار	میانگین		واریانس		بیشینه		کمینه	
		MDD	AU	MDD	AU	MDD	AU	MDD	AU
2	3	0.85%	5.39%	0.04%	1.44%	9.03%	51.07%	0.03%	0.03%
	4	0.52%	8.17%	0.01%	0.79%	6.26%	29.68%	0.02%	0.03%
	5	3.11%	9.19%	0.38%	0.76%	30.32%	34.52%	0.02%	0.02%
3	3	2.76%	5.30%	0.27%	0.37%	20.60%	20.60%	0.03%	0.03%
	4	1.77%	5.55%	0.08%	0.39%	10.88%	29.04%	0.06%	0.06%

جدول ۷: تعداد تکرارهای لازم برای دستیابی به پاسخ مدل ریاضی.

پردازنده	کار	میانگین	بیشینه	کمینه
2	3	3,481	4,868	2,204
	4	82,317	119,881	53,965
	5	2,865,682	3,843,161	1,428,459
3	3	54,237	80,390	17,547
	4	5,088,109	7,468,770	2,524,358

تخصیص بهینه منابع دست یافت. سپس بر مبنای اصول راهکارهای رایج و کارآمد MDD و AU، الگوریتمی جداگانه ای برای تعیین توالی ارائه گردید. سپس کارایی این الگوریتم در پاسخ یابی چند مساله نمونه ارزیابی شد و در نهایت اعتبار پاسخهای حاصل از آن با جوابهای قطعی حاصل از حل مدل ریاضی برای مجموعه‌ای از مسایل سنجش شد.

فهرست علائم

N	تعداد کل کارها
M	تعداد کل پردازنده ها
$Q(i)$	مقدار کل کار i
$d(i)$	موعد تحویل کار i
$p(i)$	زمان پردازش یک واحد از کار i
$S(i,j)$	زمان راه اندازی برای پردازش کار j بر روی ماشین که پیشتر کار i را پردازش نموده است (برابر S_{sim} یا S_{nsim})
$S(0,j)$	زمان راه اندازی لازم برای پردازش کار j بر روی ماشین که پیشتر بیکار بوده است
$Z(i)$	مقدار دیرکرد کار i
$C(i,k)$	زمان تکمیل کار i بر روی ماشین k
$q(i,k)$	مقدار کار i که بر روی ماشین k پردازش می شود
$x(i,j,k)$	برابر با ۱ اگر کار j پس از کار i روی ماشین k پردازش شود و در غیر اینصورت صفر است.
$x(0,j,k)$	برابر است با ۱ اگر کار j اولین کار روی ماشین k باشد و در غیر اینصورت برابر صفر است.
$x(j,0,k)$	برابر است با ۱ اگر کار j آخرین کار روی ماشین k باشد و در غیر اینصورت برابر صفر است.
CUB	عددی ثابت بزرگتر از هر زمان تکمیل ممکن

اما نکته‌ای که لازم به توجه می‌باشد آن است که اختلاف نتایج حاصل از الگوریتم‌های تفاوت اندکی با مقدار پاسخ بهینه دارد و میزان پراکندگی آنها نیز قابل اغماض است؛ به عنوان نمونه الگوریتم MDD حتی در بدترین حالت نیز بیش از ۳٪ اختلاف ندارد. این نکته از آنجهت اهمیت می‌یابد که زمان دسترسی به پاسخ قطعی و تخمینی با وجود اختلاف کم جوابها، بسیار متفاوت است. به عبارت دیگر تعداد تکرارها برای ترکیبات مختلف تعداد کار و پردازنده با افزایش تعداد کارها به طور نمایی افزایش می‌یابد. این افزایش با ورود ماشینهای جدید شدت بیشتری می‌یابد؛ بطوریکه به عنوان نمونه مشخص شد که برای برنامه‌ریزی ۶ کار بر روی ۲ ماشین به بیش از ۲۵۰,۰۰۰,۰۰۰ تکرار نیاز است که بر روی یک رایانه معمولی به معنای حدود ۴۰ ساعت زمان است.

بطور کلی، مزیت اصلی روش‌های اکتشافی که آنها را کاربردی‌تر از روش‌های قطعی می‌سازد دستیابی سریع به پاسخی تا حد قابل قبولی نزدیک به جواب بهینه است. همانگونه که از نتایج حاصله از آزمون برمی‌آید میزان دوری از پاسخ قطعی، در مقایسه با سرعت پاسخ‌یابی الگوریتم که در کسری از ثانیه جواب را ارائه می‌دهد، آنچنان زیاد نیست. بعلاوه با توجه به ابعاد مسایل واقعی، نمی‌توان روشی جز راهکارهای اکتشافی را متصور شد.

نتیجه گیری

در این مقاله مساله زمانبندی کارهای قابل تقسیم در محیط ماشین‌های موازی با در نظر گرفتن زمان آماده‌سازی وابسته به توالی مورد مطالعه و بررسی قرار گرفت. از آنجاکه تا حد اطلاع نویسندگان هیچ تحقیق مشابهی در این زمینه صورت نگرفته است، سعی شد تا با توجه به پیچیدگی مساله، رهیافت مناسب و کاربردی ارائه گردد. در این راستا دو زیر مساله تخصیص منابع و تعیین توالی مطالعه شد و با ارائه و اثبات دو قضیه، نشان داده شد که با استقلال مناسبی از توالی عملیات، می‌توان به

مراجع

- 1 - Pinedo, M. (2001). *Scheduling: Theory, Algorithms, and Systems*. Prentice-Hall Inc., Englewood Cliffs, N.J.
- 2 - Conway, R. W., Maxwell, W. L. and Miller, L. W. (2003). *Theory of Scheduling*. Dover Publications.
- 3 - Beker, K. R. (1974). *Introduction to Sequencing and Scheduling*. John Wiley & Sons Inc. West Sussex, UK.
- 4 - Tavakkoli-Moghaddam, R. and Daneshmand-Mehr, M. (2005). "A computer simulation model for job shop scheduling problems minimizing makespan." *Computers and Industrial Engineering*, Vol. 48, No. 4, PP. 811-823.

- 5 - Jolai, F. (2005). "Minimizing number of tardy jobs on a batch processing machine with incompatible job families." *European Journal of Operational Research*, Vol. 162, PP. 184-190.
- 6 - Lenstra, J. K., Rinnooy Kan, A. H. G. and Brucker, P. (1977). "Complexity of machine scheduling problems." *Annals Discrete Mathematics*, Vol. 1, PP. 343-362.
- 7 - Du, J. and Leung, J. Y. T. (1990). "Minimizing total tardiness on one machine is NP-hard." *Mathematics and Operations Researches*, Vol. 15, PP. 483-495.
- 8 - Koulamas, C., (1994). "Total tardiness Problem: review and extensions." *Operations Research*, Vol. 42, No.6, pp. 1025-1041.
- 9 - Lawler, E. L., Lenstra, J. K. and Rinnooy Kan, A.H.G. (1982). "Deterministic and stochastic scheduling." *Reidel Dordrecht*, PP. 35-37.
- 10 - Gupta, J. N. and Maykut, A. R. (1973). "Scheduling jobs on parallel processors with dynamic programming." *Decision Science*, Vol. 4, PP. 447-457.
- 11 - Elmaghraby, S. E., Park, S. H. (1974). "Scheduling jobs on a number of identical machines." *AIIE Transactions*, Vol. 6, PP. 1-13.
- 12 - Barnes, J. W. and Brennan, J. J. (1977). "An improved algorithm for Independent jobs to reduce mean finishing times." *AIIE Transactions*, Vol. 17, PP. 382-387.
- 13 - Yalaoui, F. and Chu, C. (2002). "Parallel machine scheduling to minimize total tardiness." *International Journal of Production Economics*, Vol. 76, PP. 265-279.
- 14 - Lawler, E. L. (1964). "On scheduling problems with deferral costs." *Management Science*, Vol. 11, PP. 280-288.
- 15 - Root, J.G. (1965). "Scheduling with deadlines and loss function on k parallel machines." *Management Science*, Vol. 11, PP. 460-475.
- 16 - Pritsker, A. A. B., Walters, L. J. and Wolfe, P. M. (1969). "Multi-project scheduling with limited resource: a zero-one programming approach." *Management Science*, Vol. 16, PP. 93-108.
- 17 - Montagne, J. (1969). "Sequencing with time delay costs." *Industrial Engineering Research Bulletin*, Arizona State University, Vol. 5, PP. 20-31.
- 18 - Carroll, D. C. (1965). *Heuristic sequencing of single and multiple components*. Ph.D. Dissertation, Sloan School of Management, Massachusetts Institute of Technology, MA.
- 19 - Morton, T. E., Rachamadugu, R. M. and Vepsalainen, A. (1984). "Accurate myopic heuristics for tardiness scheduling." *Working paper*, Carnegie Mellon University, Pittsburgh, Penn.
- 20 - Wilkerson, L. J. and Irwin, J. D. (1971). "An improved algorithm for scheduling independent tasks." *AIIE Transactions*, Vol. 3, PP. 239-245.
- 21 - Ho, J. C. and Chang, Y. L. (1991). "Heuristic for minimizing mean tardiness for m parallel machines." *Naval Research Logistics*, Vol. 38, PP. 367-381.
- 22 - Dogramaci, A. and Surkis, J. (1979). "Evaluation of a heuristic for scheduling independent jobs on parallel identical processors." *Management Science*, Vol. 25, PP. 1208-1216.
- 23 - Koulamas, C. (1997). "Decomposition and hybrid simulated annealing heuristics for the parallel-machine total tardiness problem" *Naval Research Logistics*, Vol. 44, PP. 109-125.
- 24 - Chen, K., Wong, J. S., Ho and J. C. (1997). "A heuristic algorithm to minimize tardiness for parallel machines." *Proceedings of ISMM/International Conference*, ISSM-ACTA Press, Anaheim, CA, USA, PP. 118-121.
- 25 - Baker, K. R. and Bertrand, J. W. (1982). "A dynamic priority rule for sequencing against due-date." *Journal of Operations Management*, Vol. 3, PP. 37-42.
- 26 - Alidaee, B. and Rosa, D. (1997). "Scheduling parallel machines to minimize total weighted and un-weighted tardiness." *Computers and Operations Research*, Vol. 24, No.8, PP. 775-788.
- 27 - Lee, Y.H. and Pinedo, M., (1997). "Scheduling jobs on parallel machines with sequence dependent setup times" *European Journal of Operational Research*, Vol. 100, PP. 464-474.

واژه‌های انگلیسی به ترتیب استفاده در متن

- | | |
|--------------------------------|-----------------------------|
| 1 - Montage's Ratio Method | 6 - High Priority Job First |
| 2 - Cost Over Time | 7 - Modified Due Date |
| 3 - Apparent Urgency | 8 - Preemption |
| 4 - Traffic Congestion Ratio | |
| 5 - Hybrid Simulated Annealing | |