مجله نخبگان علوم و مهندسی

Journal of Science and Engineering Elites

جلد ۲- شماره ۶- سال ۱۳۹۶

www.ElitesJournal.ir

افزایش کارایی ماشین‌های موازی با استفاده ازالگوریتم برنامه‌ریزی دینامیک

شراره مهاجری[۱*]، مسعود پیردستان[۲]، امیر حسین آزاد نیا[۳]

۱- دانشجوی دکتری مهندسی صنایع، باشگاه پژوهشگران جوان و نخبگان، واحد نور، دانشگاه آزاد اسلامی نور، ایران

۲- دانشجوی دکتری مهندسی صنایع، واحد نور، دانشگاه آزاد اسلامی نور، ایران

۳- مدرس و عضو هیات علمی، دانشگاه آزاد اسلامی آیت الله آملی، ایران

*mohajerishararareh@gmail.com

ارسال: دی ماه ۹۶ پذیرش: بهمن ماه ۹۶

خلاصه

هدف این تحقیق برنامه ریزی و در نتیجه بیشینه کردن مقدار کار انجام شده قبل از پیش آمدن وقفه می‌باشد. در این تحقیق یک مسئله برنامه‌ریزی n شغل غیر وابسته در m ماشین در نظر گرفته می شود. که تحت وقفه‌های غیرقابل پوشش می‌باشد. به طور کلی، شانس موفقیت در کار به همان اندازه وابسته به شکست در فرایند است که به ویژگی‌های کار وابسته می‌باشد. عموماً هر چه کار کوتاه‌تر باشد، احتمال موفقیت بیشتر خواهد بود. اگر درحین کار یک ماشین، خطایی رخ دهد، کار انجام شده نمی‌تواند به اندازه مقدار کار برنامه‌ریزی شده در ماشین باشد و در نتیجه منجر به عدم دستیابی به خروجی مورد انتظارمی شود. تحقیق پیش رو که با برنامه‌ریزی کارهای غیرقابل پیشگیری در ماشین‌های موازی در معرض وقفه‌های غیرقابل برگشت انجام می شود، شامل تخصیص مشاغل به ماشین آلات و تعیین توالی آن‌ها برای بیشینه کردن مقدار کار کامل شده ای که به وسیله آن سیتم حاصل شده‌است، می باشد. در این تحقیق به بررسی مسئله در حالتی که توزیع خرابی به صورت نمایی است، پرداخته می شود و مسئله حالت خاصی از مسائل چندجمله‌ای که قابل حل هستند، می‌باشد. هدف از این امر برنامه‌ریزی کارها در بین ماشین‌ها می‌باشد، به گونه ای که میزان کارتکمیل شده مورد انتظار بیشینه شود و یک الگوریتم برنامه‌ریزی دینامیک بهینه پیشنهاد داده شد. آزمایشات محاسباتی که از هردو نقطه نظر کیفیت و زمان محاسبات نشان دهنده عملکرد مناسب این الگوریتم در شرایط تصادفی می باشد.

کلمات کلیدی: الگوریتم، برنامه ریزی دینایک، ماشین های موازی.

۱. مقدمه

در این مقاله مسئله برنامه‌ریزی n شغل در m ماشین را در نظر می‌گیریم که شامل وقفه‌های غیرقابل پوشش می‌باشد. معمولا هر چه کار کوتاه‌تر باشد، احتمال موفقیت بیشتر است. برای فرایند شکست در این تحقیق، موردی در نظر گرفته شده است که در آن توزیع بین شکست‌ها از نوع نمایی می‌باشد. این امر مبین آن خواهد بود که احتمال موفقیت یک کار به زمان شروع آن بستگی ندارد، بلکه به طول مدت آن وابسته می باشد. تحقیقی که با مضمون برنامه‌ریزی کارهای غیرقابل پیشگیری در ماشین‌های موازی در معرض وقفه‌های غیرقابل بازیابی انجام می شود، شامل تخصیص مشاغل به ماشین آلات و تعیین توالی آن‌ها برای بیشینه کردن مقدار کار کامل شده ای که به وسیله آن سیستم حاصل شده‌است.

در این تحقیق نتایجی بیان می‌شود که برروی پیچیدگی EUISP متمرکز شده و روش‌های حل پایداری دارند. در ابتدا در صورت توزیع یکسان شکست برای ماشین‌ها مسئله می‌تواند به عنوان شکل خاصی فرمول بندی شود که تحت عنوان مسئله زمان‌بندی کاری غیرقابل اتکا(UJP) شناخته می‌شود [۱].

سپس نشان داده می شود که حتی در (m=2) ۲ماشین مشابه، EUISP یک مسئله NP-Hard می‌باشد و یک کلاس خاص از مواردی که در آن قابلیت حل موثر EUISP دارد را طبقه بندی میکنیم. بنابراین ما دو دیدگاه حل را در نظر گرفته، که درآن (i) یک الگوریتم حل دقیق بر پایه برنامه‌ریزی دینامیک و (ii) یک الگوریتم برنامه‌ریزی لیستی هستند. الگوریتم دقیق دارای پیچیدگی شبه چندجمله‌ای می‌باشد، که این نکته نشان می‌دهد که EUISP یک مسئله np-hard نمی‌باشد. بررسی‌های محاسباتی برای هر دو دیدگاه به کار برده‌شده و توسط حد بالا که می‌تواند به سرعت محاسبه شود، نشان داده خواهد شد.

۲. روش پژوهش

مسئله‌ای که در این تحقیق به آن اشاره شده ، شامل تخصیص کار به m ماشین و تعیین توالی آن می‌باشد، به صورتی که مجموع کار انجام شده مورد انتظار بیشینه شود. در نظر بگیرد که J و M به ترتیب مجموعه‌های $\{J_1, \dots, J_n\}$ و $\{M_1, \dots, M_m\}$ هستند، که در آن $|J| = n$ و $|M| = m$ اندازه هرکدام از مجموعه‌ها می‌باشد. برای انجام کار J_j، زمان پردازش p_j در هرماشین لازم می‌باشد. فرض بر این است که مشاغل به ترتیب SPT شماره گذاری شده اند. ماشین‌ها تحت شکست‌های غیرقابل بازیابی قرار می‌گیرند واگر یک ماشین در حین پردازش دچار مشکل شود، کار مرتبط به آن و همگی کارهای برنامه ریزی شده در آن ماشین از برنامه خارج خواهند شد. در این تحقیق موردی را در نظر گرفته‌ایم که در آن برای ماشین M_i زمان مابین شکست‌ها دارای توزیع نمایی با پارامتر λ_i می‌باشد، به عنوان مثال اگر M_i در زمان مشخصی فعال باشد، احتمال آنکه بعد از گذشت t واحد زمانی همچنان در حال انجام کار خود باشد توسط رابطه $e^{-\lambda_i t}$ به دست خواهد آمد. در حقیقت با توجه به توزیع نمایی که فاقد حافظه می‌باشد، این احتمال به زمان گذشته از آخرین شکست بستگی نداشته و فقط به t بستگی دارد. از این رو اگر ماشین M_i در زمانی کار J_j آغازبه کار می‌کند آماده باشد، احتمال آنکه این ماشین در حین اجرای J_j دچار شکست نشود برابر خواهد بود با $\pi_j = e^{-\lambda_i p_j}$.

π_j رااحتمال موفقیت کار J_j می‌نامیم. اگر کار J_j با موفقیت به وسیله یک ماشین پردازش شود به مقدار کار p_j دست پیدا میکنیم. با توجه به $\sigma = \{\sigma_1, \sigma_2, \dots, \sigma_m\}$ یک برنامه ریزی انجام می شود. برای مثال، تخصیص n کار به m ماشین همراه با توالی شغلی σ_i برای هر ماشین M_i $(i = 1, \dots, m)$. با داشتن زیر مجموعه‌ای از K کار که به ماشین M_i تخصیص داده شده است و داشتن پارامتر شکست λ_i، توالی داده شده σ_i برای این کارها، کاری که در hامین محل قرار دارد را مشخص کرده و میزان کار انجام شده مورد انتظار ۱ به وسیله ماشین M_i،با عنوان $EAW[\sigma_i]$ به وسیله رابطه زیر به دست خواهد آمد.

$$EAW_i[\sigma_i] = p_{\sigma_i(1)} e^{-\lambda_i p_{\sigma_i(1)}} + p_{\sigma_i(2)} e^{-\lambda_i (p_{\sigma_i(1)} + p_{\sigma_i(2)})} + \dots + p_{\sigma_i(K)} e^{-\lambda_i (p_{\sigma_i(1)} + p_{\sigma_i(2)} + \dots p_{\sigma_i(K)})} = \sum_{h=1}^{K} p_{\sigma_i(h)} e^{-\lambda_i \sum_{j=1}^{h} p_{\sigma_i(j)}}. \qquad (۱)$$

در جایی که همگی ماشین‌ها پروسه شکست یکسانی را خواهند داشت، مسئله را به عنوان EUISP*(m) بیان می شود. اگر $m = 1$ باشد، از EUISP(1) یا EUISP*(1) استفاده می شود.

۱-۲- رابطه با مسئله برنامه‌ریزی کار غیرقابل اطمینان

EUISP*(m) شکل خاصی از مسئله برنامه‌ریزی کار غیرقابل اطمینان (UJP) می‌باشد. در UJP(m)، n شغل وجود دارد که باید به m ماشین تخصیص و توالی آن مشخص گردد. هیچ زمان پردازشی مشخص نشده‌است، اما برای کار J_j یک احتمال

موفقیت مشخص π_j و یک در آمد r_j وجود دارد. همانند $EUSP^*(m)$، اگر یک کار شکست بخورد، ماشینی که کار به آن تخصیص داده دیگر قادر به انجام آن نخواهد بود. وقتی که یک مجموعه از K کار با توجه به ترتیب σ_i برای ماشین M_i مرتب شده‌اند، درآمد $Z_i(\sigma_i)$ برای M_i با استفاده از رابطه زیر بدست خواهد آمد [۲]:

$$z_i(\sigma_i) = r_{\sigma_i(1)}\pi_{\sigma_i(1)} + r_{\sigma_i(2)}\pi_{\sigma_i(1)}\pi_{\sigma_i(2)}+\cdots+r_{\sigma_i(K)}\pi_{\sigma_i(1)}\pi_{\sigma_i(2)}\cdots \pi_{\sigma_i(K)} = = \sum_{h=1}^{K} r_{\sigma_i(h)}\left(\prod_{j=1}^{h}\pi_{\sigma_i(j)}\right). \quad (۲)$$

$UJP(m)$ شامل تخصیص کارها به ماشین‌ها و تعیین توالی آن می‌شود که در نتیجه آن درآمد مورد انتظار بیشینه خواهد شد. $EUISP^*(m)$ شکل خاصی از $UJP(m)$ می‌باشد که در آن $r_j = p_j$ و $\pi_j = e^{-\lambda p_j}$ می‌باشند. زمانی‌که m برابر با یک باشد ($m = 1$)، $UJP(1)$ می‌تواند با استفاده از مرتبه غیرافزایشی نسبت Z زیر به طور موثر حل گردد، که این رابطه به صورت زیر شده‌است [۳]:

$$Z_j = \frac{\pi_j r_j}{1 - \pi_j}. \quad (۳)$$

با توجه به $EUISP^*(1)$ ، نسبت Z_j به صورت زیر درخواهد آمد:

$$Z_j = \frac{p_j e^{-\lambda p_j}}{1 - e^{-\lambda p_j}} \quad (۴)$$

برای هر $\lambda > 0$، تابع

$$f(x) = \frac{x e^{-\lambda x}}{1 - e^{-\lambda x}} \quad (۵)$$

به صورتی یکنواخت در بازه $[0, +\infty)$ کاهش خواهد یافت. در نتیجه اگر دوکار J_j و J_k به گونه‌ای باشند که $p_j < p_k$ باشد، پس $Z_j > Z_k$ خواهد بود و تئوری زیر برقرارخواهد بود:

$$\pi_1 \geq \pi_2 \geq \ldots \geq \pi_n. \quad (۶)$$

مورد UJP شرایط پذیرش را دارد اگر:

$$\pi_1 r_1 \geq \pi_2 r_2 \geq \ldots \geq \pi_n r_n. \quad (۷)$$

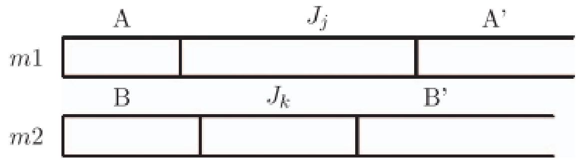
لازم به ذکر است که (۶) و (۷) هردو از رابطه زیر پیروی می کنند.

$$Z_1 \geq Z_2 \geq \ldots \geq Z_n. \quad (۸)$$

در σ در ماشین M_{m_1} مجموعه‌ی A از کارها برنامه ریزی شده است، ابتدا کار J_j و بعد از آن مجموعه کار A' برنامه‌ریزی خواهندشد. در ماشین M_{m_2} مجموعه‌ایی از کارهای B برنامه ریزی شده است که بعد از آن J_k و بعد از آن B' برنامه ریزی شده‌اند(شکل ۱). لازم به ذکر است که زمانی که J_j برنامه‌ریزی می‌شود M_{m_1} ماشینی خواهد بود که بیشترین میزان احتمال انباشته را خواهد داشت.

$$\Pi(A) > \Pi(B) \quad (۹)$$

مجموع درآمد موردانتظار برای برنامه σ با استفاده از رابطه زیر به دست خواهد آمد:

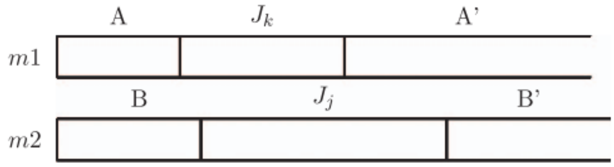


شکل ۱- برنامه σ

$$z(\sigma) = z(A) + \Pi(A)r_j\pi_j + \Pi(A)\pi_j z(A') + z(B) + \Pi(B)r_k\pi_k + \Pi(B)\pi_k z(B') \tag{۱۰}$$

حال فرض کنید که برنامه σ' که در آن کار J_k و J_j با یکدیگر جابه‌جا شده‌اند (شکل ۲). مجموع درآمد موردانتظار برنامه σ' با استفاده از رابطه زیر بدست می‌آید.

$$z(\sigma') = z(A) + \Pi(A)r_k\pi_k + \Pi(A)\pi_k z(A') + z(B) + \Pi(B)r_j\pi_j + \Pi(B)\pi_j z(B') \tag{۱۱}$$



شکل ۲- برنامه σ'

از ۱۰ و ۱۱ برمی‌آید که برنامه σ' نسبت به برنامه σ از برتری برخوردار است، برای مثال رابطه $z(\sigma') - z(\sigma) \geq 0$ در صورتی برقرار خواهد بود که اگر و تنها اگر

$$\Pi(A)(r_k\pi_k - r_j\pi_j + \pi_k z(A') - \pi_j z(A')) + \Pi(B)(r_j\pi_j - r_k\pi_k + \pi_j z(B') - \pi_k z(B')) \geq 0. \tag{۱۲}$$

$$(\Pi(A) - \Pi(B))(r_k\pi_k - r_j\pi_j) + (\Pi(A)z(A') - \Pi(B)z(B'))(\pi_k - \pi_j) < 0 \tag{۱۳}$$

از رابطه ۹ می‌دانیم که $\Pi(A) \geq \Pi(B)$ در حالی که شرایط پذیرش $r_k\pi_k \geq r_j\pi_j$ و $\pi_k \geq \pi_j$ می‌باشد. پس رابطه (۱۳) بیان کننده رابطه زیر خواهد بود:

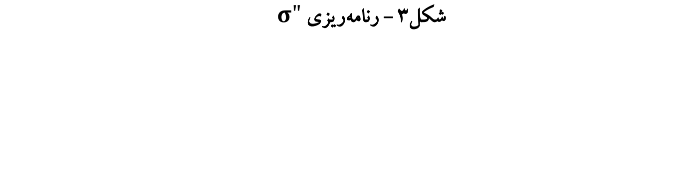
$$\Pi(A)z(A') - \Pi(B)z(B') < 0$$

و چون $\Pi(A) \geq \Pi(B)$، می‌توان گفت

$$z(A') < z(B'). \tag{۱۴}$$

برنامه جدید σ'' را در نظر بگیرید که از σ با جابه‌جایی J_jOA' در M_{m_1} با J_kOB' در M_{m_2} بدست می‌آید(شکل ۳). درآمد مورد انتظار از برنامه σ'' با استفاده از رابطه زیر بدست می‌آید

$$z(\sigma'') = z(A) + \Pi(A)r_k\pi_k + \Pi(A)\pi_k z(B') + z(B) + \Pi(B)r_j\pi_j + \Pi(B)\pi_j z(A'). \tag{۱۵}$$



شکل۳- رنامه‌ریزی σ''

از روابط ۱۰ و ۱۵ می‌توان نتیجه‌گیری کرد که "σ از σ بهتر است، برای مثال رابطه $z(\sigma'') - z(\sigma) \geq 0$ برقرار خواهد بود اگر رابطه زیر برقرار باشد:

$$\Pi(A)(r_j\pi_j - r_k\pi_k) + \Pi(A)(\pi_j z(B') - \pi_k z(A')) + \Pi(B)(r_k\pi_k - r_j\pi_j) + \Pi(B)(\pi_k z(A') - \pi_j z(B')) \geq 0,$$

و

$$(\Pi(A) - \Pi(B))(r_k\pi_k - r_j\pi_j) + (\Pi(A) - \Pi(B))(\pi_k z(B') - \pi_j z(A')) \geq 0 \quad (۱۶)$$

با تقسیم رابطه ۱۶ بر $\Pi(A) - \Pi(B) > 0,$ و دانستن این امر که $r_k\pi_k \geq r_j\pi_j$ و $\pi_k \geq \pi_j,$ برقرار هستند، از رابطه ۱۴ ($z(B') > z(A')$) خواهیم داشت که $z(\sigma'') - z(\sigma) \geq 0$. از این رو با داشتن σ، همواره می‌توان $\tilde{\sigma}$ را به دست آورد به گونه‌ای بدتر از σ نبوده و آن کار J_k با کار J_j جابه‌جا می‌شود.اگر ۱۲ رابطه برقرار باشد، $\tilde{\sigma}$ برابر خواهد بود با $\tilde{\sigma} = \sigma'$σ، در غیر این صورت $\tilde{\sigma}$ برابر خواهد بود با "σ ($\tilde{\sigma} = \sigma''$).

درنتیجه، LSA به طور صحیح نسبت به حل مسئله عمل می‌کند. در رابطه با فاز برنامه‌ریزی این امر با استفاده از فاز مرتب سازی حل می‌شود.

۲-۱ پیچیدگی EUISP*(m) برای $m \geq 2$

در این بخش به بررسی پیچیدگی EUISP*(m) می‌پردازیم. به یاد داشته باشید که برای EUISP*(m) تنها در صورتی قابل قبول است که اگر و تنها اگر رابطه زیر برقرار باشد.

$$p_1 e^{-\lambda p_1} \geq p_2 e^{-\lambda p_2} \geq \cdots \geq p_n e^{-\lambda p_n}. \quad (۱۷)$$

و

$$j < k \Leftrightarrow p_j \leq p_k \Leftrightarrow p_j e^{-\lambda p_j} \geq p_k e^{-\lambda p_k}$$

N کار وجود دارد که طول آن‌ها به صورت $p_j = a_j$ می باشد. سری مک لورن تابع $e^{-\lambda x}$ به صورت زیر می‌باشد:

$$e^{-\lambda x} = 1 - \lambda x + \frac{1}{2}\lambda^2 x^2 - \ldots \quad (۱۸)$$

به دلیل اینکه λ بسیار کوچک می‌باشد، ما توانیم تابع نمایی سری مک لورن را تقریباً برابر دو عبارت آن بدانیم یعنی $e^{-\lambda x} \simeq 1 - \lambda x.$. این امر کمک میکند تا بتوانیم میزان کار تقریبی مورد انتظار $EAW_i^{approx}[\sigma_i]$ را در ماشین M_i به صورت تقریبی به دست بیاوریم، $i = 1,2$

$$EAW_i^{approx}[\sigma_i] = \sum_{h=1}^{K} p_{\sigma_i(h)}(1 - \lambda \sum_{j=1}^{h} p_{\sigma_i(j)}). \quad (۱۹)$$

و

$$EAW^{approx}[\sigma] = EAW_1^{approx}[\sigma_1] + EAW_2^{approx}[\sigma_2] \quad (۲۰)$$

بخش (i). برنامه $\sigma = \{\sigma_1, \sigma_2\}$ را در نظر می‌گیرد، که در آن کارهای n_1 و n_2 در ماشین‌های M_1 و M_2 برنامه ریزی شده‌اند.

از رابطه (۱۹) می‌دانیم که در $i=1,2$، رابطه زیر برای ماشین M_i برقرار خواهد بود.

باتوجه به رابطه ۱۸، از آن جهت که ما تنها موردی را نگاه می‌کنیم که در آن $\lambda x < 1$ می‌باشد، مجموع عباراتی که حذف میشوند مقداری مثبت

(۲۱)

$$EAW_i^{approx}[\sigma_i] = \sum_{k=1}^{n_i} p_{\sigma_i(k)}(1 - \lambda(\sum_{j=1}^{k} p_{\sigma_i(j)}))$$

(۲۲)

$$= \sum_{k=1}^{n_i} p_{\sigma_i(k)} - \lambda \sum_{k=1}^{n_i} p_{\sigma_i(k)}^2 - \lambda \sum_{k,j:k<j} p_{\sigma_i(k)} p_{\sigma_i(j)} = \sum_{k=1}^{n_i} p_{\sigma_i(k)} - \lambda \sum_{k=1}^{n_i} p_{\sigma_i(k)}^2 - \lambda \frac{(\sum_{k=1}^{n_i} p_{\sigma_i(k)})^2 - \sum_{k=1}^{n_i} p_{\sigma_i(k)}^2}{2} = \sum_{k=1}^{n_i} p_{\sigma_i(k)} - \frac{1}{2}\lambda \sum_{k=1}^{n_i} p_{\sigma_i(k)}^2 - \frac{1}{2}(\sum_{k=1}^{n_i} p_{\sigma_i(k)})^2$$

$$EAW_i[\sigma_i] > EAW_i^{approx}[\sigma_i].$$ (۲۳)

از رابطه ۲۲ داریم که مجموع کارمورد انتظار تقریبی با استفاده از رابطه زیر به دست می آید:

$$EAW^{approx}[\sigma] = EAW_1^{approx}[\sigma_1] + EAW_2^{approx}[\sigma_2] = = \sum_{k=1} p_{\sigma_1(k)} - \frac{1}{2}\lambda \sum_{k=1}^{n_1} p_{\sigma_1(k)}^2 - \frac{1}{2}\lambda\left(\sum_{k=1}^{n_1} p_{\sigma_1(k)}\right)^2 + \sum_{k=1}^{n_2} p_{\sigma_2(k)} - \frac{1}{2}\lambda \sum_{k=1}^{n_2} p_{\sigma_2(k)}^2 - \frac{1}{2}\lambda\left(\sum_{k=1}^{n_2} p_{\sigma_2(k)}\right)^2 = \sum_{k=1}^{n} p_k - \frac{1}{2}\lambda \sum_{k=1}^{n} p_k^2 - \frac{1}{2}\lambda\left(\sum_{k=1}^{n_1} p_{\sigma_1(k)}\right)^2 - \frac{1}{2}\lambda\left(\sum_{k=1}^{n_2} p_{\sigma_2(k)}\right)^2 = Q - \frac{1}{2}\lambda\left[\left(\sum_{k=1}^{n_1} p_{\sigma_1(k)}\right)^2 + \left(\sum_{k=1}^{n_2} p_{\sigma_2(k)}\right)^2\right].$$ (۲۴)

با توجه به اینکه $Q = \sum_{k=1}^{n} p_k - \frac{1}{2}\lambda \sum_{k=1}^{n} p_k^2$ به بخش کارهای بین دو ماشین بستگی ندارد. از این رو در رابطه ۲۴ مقدار $EAW^{approx}[\sigma]$ زمانی بیشینه خواهد بود که مقدار رابطه زیر کمینه باشد:

$$\left(\sum_{k=1}^{n_1} p_{\sigma_1(k)}\right)^2 + \left(\sum_{k=1}^{n_2} p_{\sigma_2(k)}\right)^2$$ (۲۵)

زمانی که هردو جمع برابر باشند یک حد پایین برای برای مقادیر اعمال می‌شود. برای مثال

$$\sum_{k=1}^{n_1} p_{\sigma_1(k)} = \sum_{k=1}^{n_2} p_{\sigma_2(k)} = W/2.$$

اگر یک برنامه σ^* وجود داشته باشد، در نتیجه مقدار $EAW^{approx}[\sigma^*]$ برابر خواهد بود با $Q + W^2/2$.

از روابط پایه‌ای لاگرانژ می‌دانیم که برای هر x در عبارات باقیمانده سری رابطه مک لورن همواره $\xi \in [0, x]$ وجود دارد که رابطه به صورت زیر در می‌آید:

$$e^{-\lambda x} = 1 - \lambda x + \frac{1}{2}\lambda^2 x^2 e^{-\lambda \xi}$$ (۲۶)

براساس روابط ۱ تا ۲۶ در هر برنامه σ_i می‌توان رابطه $EAW_i[\sigma_i]$ را به صورت زیر تعریف کرد:

$$EAW_i[\sigma_i] = \sum_{k=1}^{n_i} p_{\sigma_i(k)}\left(1 - \lambda \sum_{j=1}^{k} p_{\sigma_i(j)} + \frac{1}{2}\lambda^2\left(\sum_{j=1}^{k} p_{\sigma_i(j)}\right)^2 e^{-\lambda \xi_k}\right)$$

در اینجا برای هر k، $0 \le \xi_k \le \sum_{j=1}^{k} p_{\sigma_i(j)}$ بوده و از رابطه ۲۱ داریم:

$$EAW_i[\sigma_i] = EAW_i^{approx}[\sigma_i] + \frac{1}{2}\lambda^2 \sum_{k=1}^{n_i} p_{\sigma_i(k)} \left(\sum_{j=1}^{k} p_{\sigma_i(j)} \right)^2 e^{-\lambda \xi_k}$$
$$= EAW_i^{approx}[\sigma_i] + R_i. \quad (۲۷)$$

با قرار دادن مقدار 2^{-H} به جای λ و مشاهده این امر که با جایگزین کردن صفر به جای ξ_k و همگی p_jها با W برای R_i رابطه زیر برقرار است:

$$R_i < \frac{1}{2} 2^{-2H} W^3 n_i^2, \quad (۲۸)$$

در نتیجه با افزودن $EAW_1^{approx}[\sigma_1]$ و $EAW_2^{approx}[\sigma_2]$ برای هر برنامه σ خواهیم داشت:

$$EAW[\sigma] \le EAW^{approx}[\sigma] + \frac{1}{2} 2^{-2H} W^3 (n_1^2 + n_2^2) < EAW^{approx}[\sigma] + \frac{1}{2} 2^{-2H} W^3 n^2. \quad (۲۹)$$

رابطه زیر را درون عبارت جای می‌دهیم:

$$H = 1 + \lceil \log_2(n^2 W^3) \rceil, \quad (۳۰)$$

با استفاده از عبارات ۲۸ و ۳۰ این نتیجه گرفته می شود که

$$R_1 + R_2 < 2^{-(H+1)}$$

برای هر برنامه σ داریم:

$$EAW[\sigma] - EAW^{approx}[\sigma] < 2^{-(H+1)}. \quad (۳۱)$$

از طرف دیگر چون همگی مقادیر p_j اعداد صحیح مثبت هستند، درصورتی که σ^* یک برنامه بخشی باشد، برای هر برنامه دیگر $\tilde{\sigma} \neq \sigma^*$ مقادیر (۲۵) حداقل برابراست با

$$\left(\frac{W}{2} + 1\right)^2 + \left(\frac{W}{2} - 1\right)^2 = \frac{W^2}{2} + 2$$

از رابطه (۲۴) به یاد داریم که در آن مقدار λ را با 2^{-H} جایگزین کردیم، در نتیجه:

$$EAW^{approx}[\sigma^*] - EAW^{approx}[\tilde{\sigma}] > 2^{-H} \quad (۳۲)$$

از روابط (۳۱) و (۳۲) داریم:

$$EAW[\tilde{\sigma}] < EAW^{approx}[\tilde{\sigma}] + 2^{-(H+1)} < EAW^{approx}[\sigma^*] - 2^{-H} + 2^{-(H+1)} < EAW^{approx}[\sigma^*], \quad (۳۳)$$

و از رابطه ۲۳ خواهیم داشت:

$$EAW[\sigma^*] > EAW^{approx}[\sigma^*], \quad (۳۴)$$

بنابراین برای هر برنامه ؛ $\tilde{\sigma} \neq \sigma^*$ داریم:

$$EAW[\tilde{\sigma}] < EAW[\sigma^*],$$

اگر σ^* یک برنامه بخشی باشد، این امر بهینه‌ترین حالت برای $EAW[\sigma]$. خواهد بود. در نتیجه جواب بهینه برای EUISP*(2) برنامه بخشی می‌باشد.

این نتایج نشان می‌دهند که EUISP*(m) (و همچنین EUISP(m)) رای $m \ge 2$ از نوع np سخت هست. با استفاده از الگوریتم برنامه‌ریزی دینامیک نسبت به حل EUISP(m) اقدام میکنیم.

۲-۲ الگوریتم شبه جمله‌ای برای EUISP(m)

در این بخش یک الگوریتم برنامه‌ریزی دینامیک برای EUISP(m)تعیین شده که دارای پیچیدگی شبه جمله‌ای برای مقادیر ثابت m است.

مجموعه $F(j,\ t_1,\ t_2,\ldots,t_m)$ را حداکثر مجموع کار مورد انتظار در یک مورد از EUISP(m) در نظر گرفته که محدود به j کار اول بوده و زمان تکمیل ماشین‌های $M_1, M_2, \ldots, M_m$ به ترتیب برابر است با $t_1, t_2, \ldots, t_m$.در نتیجه فرمول زیر برقرار می باشد [4].:

$$F(j,\ t_1,\ t_2,\ldots,t_m) = \max\left\{\begin{matrix} F(j-1,\ t_1-p_j,\ t_2,\ldots,t_m) + p_j e^{-\lambda_1 t_1} \\ F(j-1,\ t_1,\ t_2-p_j,\ldots,t_m) + p_j e^{-\lambda_2 t_2} \\ \cdots \\ F(j-1,\ t_1,\ t_2,\ldots,t_m-p_j) + p_j e^{-\lambda_m t_m} \end{matrix}\right\} \quad (۳۵)$$

فرمول (۳۵) برای تخصیص کار j می‌باشد. به دلیل اینکه در هرماشین کارها به صورت غیرافزایشی نسبت به Z مرتب شده‌اند. آخرین کار j را در آخرین محل در هر ماشین برنامه‌ریزی خواهد کرد. از این رو کار j هم به M_1 با زمان شروع $t_1 - p_j$ تخصیص میابد، یا به M_2 با زمان شروع $t_2 - p_j$ و غیره تخصیص خواهد یافت.

فرمول (۳۵) باید به صورت زیر بازنویسی می شود:

$$F(j,\ t_1,\ t_2,\ldots,t_m) = 0 \quad \text{if } j = 0 \quad (۳۶)$$

$$F(j,\ t_1,\ t_2,\ldots,t_m) = -\infty \quad \text{if} \sum_{k=1}^{j} p_k > \sum_{i=1}^{m} t_i \quad (۳۷)$$

$$F(j,\ t_1,\ t_2,\ldots,t_i - p_j,\ldots,t_m) = -\infty \quad \text{if} p_j > t_i, \quad \text{for } i = 1,\ldots,m \quad (۳۸)$$

$$F(j,\ 0,\ 0,\ldots,t_i,\ldots,0) = e^{-\lambda_i t_i} \quad \text{if} \sum_{k=1}^{j} p_k \le t_i, \quad \text{for } i = 1,\ldots,m. \quad (۳۹)$$

بنابراین بهینه‌ترین حل برای EUISP(m0 با استفاده از رابطه زیر مشخص می شود.

$$\max\{F(n,\ t_1,\ t_2,\ldots,t_m){:}\ 0 \le t_i \le \sum_{i=1} p_j i = 1,\ldots,m\}. \quad (۴۰)$$

هر مقدار $F(j,\ t_1,\ t_2,\ldots,t_m)$ می‌تواند در زمان O(m) و به وسیله رابطه ۳۵ محاسبه گردد.

۳-۲ حد بالایی برای EUISP(m)

در این بخش یک حد ترکیبی برای EUISP(m) ارائه شده‌است. این حد برای ارزیابی عملکرد الگوریتم ابتکاری LSA به کار برده می شود.

یک کار را در نظر بگیرید، که طول آن p بوده و برروی یک ماشین با نرخ شکست λ انجام می شود. مقدار کار مورد انتظار برابر خواهد بود با:

$$pe^{-\lambda p}. \quad (۴۱)$$

حال فرض کنید که ما کار را به دو دسته با طول‌های p_1 و p_2 تقسیم می‌کنیم که در آن‌ها $p_1 < p_2, p_1 + p_2 = p$ می‌باشد. مقدار کار مورد انتظار انجام شده به صورت زیر خواهد بود.

$$p_1 e^{-\lambda p_1} + p_2 e^{-\lambda p} \quad (۴۲)$$

که به وضوح بزرگتر از رابطه ۴۱ می باشد. برای میزان کارانجا شده مدنظر خواهیم داشت:

$$\int_0^p e^{-\lambda x}dx = \frac{1 - e^{-\lambda p}}{\lambda}. \tag{۴۳}$$

با داشتن یک مورد از $\mathrm{EUISP(m)}$، جواب بهینه برای مسئله $\mathrm{EUISP_R(m)}$ مقدار زیر را خواهد داشت:

$$UB = \sum_{i=1}^{m} \frac{1 - e^{-\lambda_i a_i^*}}{\lambda_i}, \tag{۴۴}$$

که

$$a_1^* = \frac{P}{\lambda_1 \sum_{i=1}^{m} \left(\frac{1}{\lambda_i}\right)} \tag{۴۵}$$

و

$$a_i^* = \frac{\lambda_1}{\lambda_i} a_1^* \quad \forall\ i = 2, \ldots, m. \tag{۴۶}$$

در $\mathrm{EUISP_R(m)}$ همگی کارها اندازه یکسانی دارند، زیرا که مسئله $\mathrm{EUISP_R(m)}$ از تقسیم کارهای $\mathrm{EUISP(m)}$ به بینهایت میکروکار به دست می‌آید. با تخصیص یک مقدار مشخص کار با مجموع زمان پردازش a_i به ماشین M_i، راه حل زیر برای $\mathrm{EUISP_R(m)}$به دست می آید:

$$UB = \sum_{i=1}^{m} \frac{1 - e^{-\lambda_i a_i}}{\lambda_i} \tag{۴۷}$$

در حالی که این رابطه وضعیت زیر را برقرار می‌کند:

$$\sum_{i=1}^{m} a_i = P. \tag{۴۸}$$

از این رو با یافتن مقدار a_i که رابطه ۴۷ را نسبت به رابطه ۴۸ بیشینه می‌کند می‌توان $\mathrm{EUISP_R(m)}$ را حل کرد. تابع لاگرانژ بدین صورت بیان خواهد شد:

$$L = \sum_{i=1}^{m} \frac{1 - e^{-\lambda_i a_i}}{\lambda_i} - \mu\left(\sum_{i=1}^{m} a_i - P\right) \tag{۴۹}$$

و در نتیجه ما برای مقادیر بهینه $a_i^*,\ i = 1, \ldots, m$ خواهیم داشت:

$$\frac{\partial L}{\partial a_i} = e^{-\lambda_i a_i^*} - \mu = 0 \tag{۵۰}$$

زمانی این رابطه برقرار می باشد که برای همه $i = 1, \ldots, m$رابطه زیر برقرار باشد:

$$e^{-\lambda_i a_i^*} = \mu \tag{۵۱}$$

درنتیجه با استفاده از رابطه ۴۸ داریم:

$$a_1^* = \frac{P}{\lambda_1 \sum_{i=1}^{m} \left(\frac{1}{\lambda_i}\right)} \tag{۵۲}$$

و از رابطه (۵۱) برای $i = 2, \ldots, m$ خواهیم داشت:

$$a_i^* = \frac{\lambda_1}{\lambda_i} a_1^*, \tag{۵۳}$$

۳. نتایج محاسباتی

برای بررسی عملکرد الگوریتم ابتکاری LSA یک مجموعه از موارد اتفاقی وآزمایشات برای هردو دسته مسائل EUISP که درآن‌ها ماشین‌ها مقادیر متفاوتی از λ_i دارندو مسائل *EUISP که در آن برای همه $i=1,...,m$، $\lambda_i = \lambda$ می‌باشد، صورت گرفت . به طور کلی دوگروه از موارد در نظر گرفته شد. در حالی که در موارد بزرگ LSA با حدبالایی بیان شده است در موارد کوچک، LSA با جواب بهینه بدست آمده از الگوریتم حل بهینه دقیق مقایسه شده‌است.

جدول ۱- پارامترهای آزمایشی برای موارد بزرگ

$m \in$	$\{2, 5, 10, 15\}$	
$n \in$	$\{25, 50, 100, 200\}$	
$p_j \sim$	$\{U[1-20], U[1-50], U[1-100]\}$	
Identical machines	$\lambda = 0.001$ and $\lambda = 0.0001$	
Non-identical machines	Low-value	High-value
Low-difference	$\lambda_i \in [0.0004, 0.0006]$	$\lambda_i \in [0.002, 0.003]$
High-difference	$\lambda_i \in [0.0001, 0.001]$	$\lambda_i \in [0.0005, 0.005]$

Identical machines ماشین‌های مشابه

Non-identical machines ماشین‌های غیر مشابه

Low- difference تفاوت کم

High difference تفاوت زیاد

موارد کوچک با قرار دادن m=2 و تغییر تعداد n کارها و مقادیر λ_i ($i=1,2$).در نظرگرفتیم که n برابر است با ۲۵، ۵۰ و ۷۵، در حالی که زمان پردازش از یک توزیع نرمال برخوردار هست، برای مثال $p_j \sim U[1-20], U[1-50], U[1-100]$. در مواری که ماشین‌ها یکسان باشد (* EUISP)، ما مقدار $1/\lambda$ را به عنوان مثال برابر با ۱۶۸ ساعت گرفته که منجر به یک زمان شکست متوسط یک هفته می شود برای موارد کلی EUISP، دو سناریو را در نظر می گیریم ، دریکی از این سناریوها λ_1 و λ_2 به ترتیب برای با ۰.۰۰۶ و ۰.۰۰۵ هستند که نشانگر اختلاف پایین است و در سناریوی دیگر λ_1 و λ_2 هر دو برابر ۰.۰۰۶ و ۰.۰۰۳ هستند . در موارد کوچک با در نظر گرفتن سه مقدار کوچک n، سه پهنای متفاوت توزیع یکپارچه و سه سناریو برای λ_i، ۲۷ مشخصه آزمایشی می‌دهد. برای هرکدام از این ۲۷ مشخصه، ۲۰ مورد به صورت اتفاقی تولید شده است.

موارد بزرگ با استفاده از تغییر m، n تولید شده‌است، پهنای توزیع زمان‌های پردازش و مقادیر λ_i($i=1,...,m$)،در جدول ۱ ارائه شده‌است. به طور کلی، سناریو های ماشین برای مقادیر کم و زیاد λ_i و تفاوت کم و زیاد بین آن‌ها در نظر گرفتیم که شامل دو سناریوی ماشین ، ۴ سناریو ماشین غیرهمسان می‌شود. در سناریوهای ماشین غیرهمسان، λ_i با فواصل مشخص در فواصل برابری که در جدول ۱ تعیین شده‌اند. برای مواردی که در سناریو مقادیر کم، اختلاف بالا که m آن‌ها برابر ۱۰ بوده و $\lambda_i \in [0.0001, 0.001]$ هست، ماشین دارای مقادیر λ_i برابر با ۰.۰۰۰۱، ۰.۰۰۰۲، ...، ۰.۰۰۱ می‌باشد. با درنظر گرفتن چهار مقدار n، ۴ مقدار برای m، ۳ توزیع یکسان و ۶ ماشین ۲۸۸ مشخصه آزمایشی بدست می‌آید. برای هرکدام از این مشخصه‌ها ۲۰ مورد تصادفی در نظر گرفته‌شد، که مجموع آن‌ها برابر ۵۷۶۰ مورد می شود.

LSA و الگوریتم برنامه‌ریزی دینامیک در برنامه C کدنویسی شده‌و در تمام موارد زمان محاسباتی لازم برای LSA قابل صرف نظر می‌باشد زمان مورد نیاز الگوریتم برنامه ریزی دینامیک در موارد کوچک از چندثانیه تا ۳۰ دقیقه (سناریوهای اختلاف بالا، n=75 و $p_j \sim U[1-100]$ تغییر می کند. برای موارد کوچک، میانگین درصد شکاف بین جواب بهینه Z^* که به وسیله الگوریتم

برنامه‌ریزی دینامیک به دست آمده و جواب Z_{LSA} که به وسیله LSA بدست آمده است. هرچه تفاوت بین مقادیر λ_i بیشتر باشد، شکاف بیشتر خواهد بود اما در اغلب موارد این شکاف کمتر از ۰.۵٪ می‌باشد. همچنین مواردی که کارهای آن بیشتر است شکاف کمتری را دارند . این امر توسط این اصل توضیح می دهد که کیفیت یک برنامه‌ریزی به شدت به کارهایی که در ابتدا برای هرماشین برنامه‌ریزی شده وابسته هستند و با رشد n الگوریتم LSA شانس بیشتری برای انتخاب یک زیرمجموعه از کارها در هر ماشین دارد.

برای هر ماشین با توزیع یکنواخت با پهنای n، شکاف میانگین درصد شکاف بین ۲۰ مورد هست که در بین حد بالا (UB) ارائه شده به وسیله رابطه ۴۴ و مقدار حل و مقدار حل z_{LSA} بدست آمده از LSA و g_{min} و g_{max}به ترتیب کمترین وبیشترین درصد شکاف می‌باشد. هرچه مقدار λ_i کوچکتر باشد شکاف نیز کوچکتر خواهد بود. سخت ترین موارد آنهایی هستند که دارای بیشترین پهنا در توزیع یکپارچه و کمترین نسبت n/m می‌باشد. در *EUISP، برای مقدار $\lambda = 0.0001$ بیشترین مقدار میانگین شکاف ۰.۳۹ می‌باشد، که برای m=15، n=25 و $U[1, 100]$برقرار می‌باشد در حالی که زمانی که شکست کارها احتمال آن ۱۰ برابر هست این امر به ۳.۸ می رسد. با افزایش λ، Z_{LSA} به صورت نمایی کاهش میابد. نتایج بدست آمده در این تحقیق نشان می‌دهد که سخت ترین سناریوها آن هایی هستند که مقدار λ آآن بیشتر است. اندازه λ_i اصلی ترین فاکتوریست که شکاف را تحت تاثیر قرار می‌دهد، میانگین شکاف در همگی موارد برای هر λ_i تعریف شده برای سناریو گزارش داده شده‌است. و میانگین فاصله آن در حدود ۰.۲٪ متغییر بوده ، در جایی که ماشین‌ها مشابه هستند و $\lambda = 0.0001$ می‌باشد، تا حدود ۳.۸٪ که برای سناریوهایی با اختلاف کم که λ_i آن در بازه $[0.002, 0.003]$قرار دارد. در ادامه سخت ترین سناریو قرار دارد که میانگین شکاف آن برابر ۸و۸۸٪ درصد و بزرگترین شکاف آن برابر ۹.۵٪ درصد است. از بین سناریوها، به طور میانگین بیشترین مقدار λ_i آن‌هایی هستندکه در بازه $[0.002, 0.003]$ قرار می گیرند.

برای سناریوهایی با مقادیر کم اینگونه تأثیرات خیلی بزرگ نیستند. در سناریوهایی با مقدار کم بیشترین تأثیر در سناریوهایی با اختلاف کم مشاهده می‌شود که برای $m = 15$ و $n = 25$ رخ خواهد داد. در این صورت فاصله شکاف ۰.۳۹٪ برای $U(1, 20)$ شروع شده و تا ۱.۸۸٪ برای $U(1, 100)$ ادامه خواهد یافت. البته در سناریوهای اختلاف کم و زیاد زمانی از وضوح بیشتری برخوردار است که شکاف برای زمان پردازش کوچک ۱.۹۲٪ می‌باشد و تا ۸.۸۴٪ برای زمان‌های پردازش بالا ادامه خواهد یافت.

برای اکثر مشخصه‌های آزمایشی ، شکافی که برای LSA تولید شده کمتر از ۲٪ بوده، که ارزیابی عملکرد مناسب الگوریتم را نشان می‌دهد. به طور کلی LSA در برابر تعداد کارها مقاوم بوده اما عملکرد آن با افزایش تعداد ماشین‌ها ، بیشترین رنج زمان پردازش و مقدار λ_i متغیر خواهد بود.

۴. بحث و نتیجه گیری

در این تحقیق نتایجی بیان شد که برروی پیچیدگی EUISP متمرکز شده و روش‌های حل پایداری داشتند و نشان داده شد که در (m=2) ۲ماشین مشابه، EUISP یک مسئله NP-Hard می‌باشد و یک کلاس خاص از مواردی که در آن قابلیت حل موثر EUISP دارد طبقه بندی شدند. هدف از این امر برنامه‌ریزی کارها در بین ماشین‌ها می‌باشد، به صورتی که میزان کارتکمیل شده مورد انتظار بیشینه گشته و یک الگوریتم برنامه‌ریزی دینامیک بهینه پیشنهاد داده شد. دو دیدگاه حل در نظر گرفته شد ه که درآن (i) یک الگوریتم حل دقیق بر پایه برنامه‌ریزی دینامیک و (ii) یک الگوریتم برنامه‌ریزی لیستی است. الگوریتم دقیق دارای پیچیدگی شبه چندجمله‌ای می‌باشد، که این نکته نشان می‌دهد که EUISP یک مسئله np-hard نمی‌باشد. بررسی‌های محاسباتی برای هر دو دیدگاه به کار برده‌شده و توسط حد بالا که می‌تواند به سرعت محاسبه شود، نشان داده شد. . آزمایشات

محاسباتی که از هردو نقطه نظر کیفیت و زمان محاسبات نشان دهنده عملکرد مناسب الگوریتم برنامه‌ریزی لیستی در شرایط تصادفی بود. تحقیقات بیشتر در این زمینه می‌تواند گسترش ابعاد مساله باشد به عنوان مثال چندین دسته با یکدیگر متصل شده‌اند تا با یکدیگر در مواقعی که یکی بیش از حد بار دارد و یا دیگر بارکاری خیلی کمی دارد، به طور داخلی همکاری داشته باشد. مقدار مشخصی از کار در اینگونه از ساختار در حال پیشرفت خواهد بود[5]. اما دیدگاه مدل‌سازی حال حاضر شکست ماشین‌ها را رد می‌کند. با یکپارچه سازی مدل‌های برنامه ریزی چند عامله با مدل‌های EUISPبه نظر می‌رسد که یکی از مسائل چالش برانگیز در آینده باشد. پیشنهاد دیگر، بررسی مواردی است در آن احتمال موفقیت به زمان شروع وابسته است. برای مثال اگر یک دستگاه خنک کننده روغن برای مدت زمان زیادی کار می‌کند، دمای آن افزایش پیدا میکند و از این رو احتمال شکست در حین انجام کار افزایش میابد..

۵. مراجع

1. Agnetis A, Detti P, Sodhi MS, Pranzo M. Sequencing unreliable jobs on parallel machines. J Sched 2009;12(1):45–54.

2. Agnetis A, Detti P, Pranzo M. The list scheduling algorithm for scheduling unreliable jobs on two parallel machines. Discret Appl Math 2014;165:2–11.

3. Mitten LG. An analytic solution to the least cost testing sequence problem. J Ind Eng 1960;11:17.

4. Benoit A, Robert Y, Rosenberg A, Vivien F. Static strategies forworksharing with unrecoverable interruptions Syst 2013;53:386–423.. Theory Comput

5. Cohen J, Cordeiro D, Trystram D, Wagner F. Coordination mechanisms for selfish multi-organization scheduling. In: Proceedings of the 18th annual IEEE conference on high performance computing (HiPC), IEEE Press; 2011, p. 1–9.