

تاثیر تعداد وظایف مختلف بر زمانبندی کارها در محیط محاسبات ابری

محمد بهروزیان نژاد^{*}

۱- گروه کامپیوتر، واحد امیدیه، دانشگاه آزاد اسلامی، امیدیه، ایران

چکیده

مدیریت منابع و زمانبندی وظایف در کنار امنیت، قابلیت اطمینان و حفظ اعتماد مشترکین یکی از چالش‌های مهم در زمینه رایانش ابری می‌باشد که می‌تواند روی سایر مسائل نیز تاثیرگذار باشد. زمان بندی وظایف تکنیکی برای توزیع عادلانه منابع مابین مشتریان است، که به منظور دستیابی به بهره‌وری بهینه منابع با کمترین زمان پاسخ است. امروزه کارهای بسیاری در این حوزه صورت گرفته است و الگوریتم‌های متنوعی به منظور بهبود زمان بندی وظایف در محیط رایانش ابر ارائه شده است. یکی از رویکردهای مطرح در این حوزه زمانبندی آگاه از توازن بار می‌باشد. در این مقاله یک راهکار مبتنی بر الگوریتم‌های بهینه‌سازی برای زمانبندی وظایف در محیط‌های ابری پیشنهاد گردیده است. برای این منظور از الگوریتم بهینه‌سازی ازدحام ذرات استفاده گردیده که در آن با بهره‌گیری از عملگرهای الگوریتم بهینه‌سازی تکامل تفاضلی سعی شده بر محدودیت‌های الگوریتم بهینه‌سازی ازدحام ذرات غلبه گردد. ارزیابی روش پیشنهادی در زمانبندی وظایف محیط‌های ابری کارایی در سه معیار ارزیابی زمان اجرای کل، ضریب عدم تعادل بار و انرژی مصرفی ماشین‌های فیزیکی براساس تاثیر تعداد وظایف مختلف مورد بررسی قرار گرفته است.

کلمات کلیدی: محاسبات ابری؛ بهینه‌سازی، تکامل تفاضلی، ازدحام ذرات.

۱. مقدمه

با پیشرفت سخت‌افزارها و سپس سیستم‌عامل‌ها و در ادامه آن نرم‌افزارها، درخواست سرویس‌های بیشتر و سرعت و قدرت بالاتر هم افزایش یافت و این وضعیت به‌جایی رسیده که کاربران بدون سخت‌افزار مناسب نمی‌توانند نرم‌افزار دلخواه خود را اجرا نمایند. با تولید و ایجاد نسخه‌های بالاتر و نرم‌افزارهای مختلف به صورتی تولید و ایجاد می‌شود که توانایی اینکه سخت‌افزار مربوطه هم به همان سرعت تغییر یابد برای کاربران به دلیل هزینه بیش از اندازه امکان‌پذیر نخواهد بود. از این رو ابرها ایجاد گردید تا نرم‌افزار و سرویس‌ها و غیره بر روی آن‌ها فعال گردیده و کاربران با پرداخت هزینه اندک و بدون نگرانی در مورد از دست دادن اطلاعات و خرابی‌های سخت‌افزاری بتوانند از سرویس خود استفاده نمایند. از طرفی ابرها نیازمند نرم‌افزارهایی برای کنترل منابع و سرویس‌ها و سخت‌افزارهای گوناگون دیگر درخواست‌های کاربران هستند که این مقوله به قسمت‌های مختلفی تقسیم شده است که یکی از موارد زمانبندی کار[†] در ابرها می‌باشد. در این مقاله طراحی سیستمی با کمترین هزینه و بیشترین بازدهی در زمان تقسیم کارهای مختلف به ابرهای مورد توجه می‌باشد [1-5].

بحث زمانبندی کار در سیستم‌های عامل یکی از بحث‌های مهم بوده و خواهد بود؛ زیرا راهکاری که بتواند با کمترین زمان بهینه‌ترین روش را پیاده‌سازی نماید همیشه مورد توجه بوده و هست. این مبحث در ابرها هم بسیار پررنگ‌تر ظاهر شده

* Corresponding author: Mohammad Behrouzian Nejad

Email: Mo.Behrouzian@iaau.ac.ir

† Task Scheduling

است، چرا که در اینجا کارها از چندین کاربر و حتی در موقعیت‌های جغرافیایی متفاوت با درخواست‌های متفاوت ارسال می‌گردد و این درخواست‌ها را باید به گونه‌ای مدیریت نمود که، هر یک دارای سرویس‌های مختلفی هستند را بررسی و در بهینه‌ترین زمان پاسخ دهد. محاسبات ابری می‌توانند کارهای پیچیده را در مدت زمان کوتاه اجرا کنند و کارایی سخت افزار را بهبود ببخشند، برای اینکه محاسبات ابری به نحو احسن کار خودش را انجام دهد باید بهترین استراتژی برای زمان بندی کارها انجام شود تا سبب بهبود انجام عملیات در حوزه های مختلف کسب و کار در بستر مناسبی برای رشد و تعالی کشور فراهم نماید.

۲. زمانبندی وظایف

در محیط محاسبات ابری، زمانبندی وظیفه و تخصیص منبع، توسط ارائه‌دهندگان خدمات از طریق فناوری مجازی مدیریت می‌شود. آنها برای مخفی و کامل کردن وظایف کاربران به صورت شفاف استفاده می‌شوند. زمانبندی وظیفه به دلیل شفافیت و انعطاف پذیری سیستم محاسبات ابری و نیازهای مختلف برای منابع، پیچیده‌تر نیز می‌شود. استراتژی‌های زمانبندی وظیفه بر عدالت یا بهره‌وری منابع تمرکز دارند که هزینه زمان، فضا و توان عملیاتی افزایش و کیفیت سرویس در محاسبات ابری بهبود خواهد داشت. زمانبندی انجام وظایف، پردازش مناسب وظایف منابع است بصورتی که وظایف بتوانند بر اساس معیارهای مورد نظر اجرا شوند. اهداف زمانبندی کارآمد برای کاربردهای مختلف، متفاوت است [6]. ویژگی‌های زمانبندی وظیفه در ابر محیط محاسبات به شرح زیر است.

- **زمانبندی وظیفه یک پلت فرم واحد و یکپارچه منابع تهیه می‌کند:** ما فرض می‌کردیم که محاسبات ابری با استفاده از فناوری‌های مجازی، منابع فیزیکی (همه نوع میزبان، ایستگاه‌های کاری و یا حتی PC و غیره) را به عنوان یک استخر منابع واحد، محافظ ناهمگن و عرضه برای استفاده بالا قرار داده است. این در حالی است که به طور عمده در تعداد زیادی از رایانه‌های توزیع شده، پخش شده‌اند و منابع در قالب یک مرکز داده عرضه می‌شوند.
- **زمانبندی وظیفه، عمومی و متمرکز شده است:** محاسبات ابری یک مدل محاسباتی است که منابع متمرکز را توسط سرویس منعکسی، به چند برنامه توزیع شده تأمین می‌کند و این به کارگیری معکوس می‌تواند روش‌های ناهمگن اجرای عملیاتی را ساده‌تر توسعه دهد؛ بنابراین، با فناوری مجازی سازی و خدمات معکوس، زمانبندی وظیفه محاسبات ابری، یک زمانبندی متمرکز جهانی به دست می‌آورد.
- **هرگره در ابر مستقل است:** در محیط محاسبات ابری، زمانبندی داخلی هر گره ابر مستقل است و زمانبندیها در ابر هیچ دخالتی باسیاست زمانبندی گرهای دیگر نخواهند داشت.
- **مقیاس پذیری زمانبندی وظیفه:** مقیاس از تأمین منابع ارائه‌دهنده ابر ممکن است در مراحل اولیه محدود باشد. علاوه بر این، انواع منابع محاسباتی و اندازه منابع مجازی انتزاعی ممکن است بزرگ شوند و تقاضای برنامه افزایش یابد. در ابر، زمانبندی وظیفه باید ویژگی‌های مقیاس‌پذیری داشته باشد، به طوری که توان عملیاتی زمانبندی وظیفه در ابر، کم نشود.
- **زمانبندی وظیفه می‌تواند به صورت پویای خود تطبیقی باشد:** گسترش و کاهش برنامه‌های کاربردی در ابر ممکن است بسته به نیاز لازم باشد. منابع محاسباتی مجازی در سیستم ابر همچنین ممکن است در همان زمان

گسترش یابند و یا کوچک شوند. منابع همواره در حال تغییرند، برخی منابع ممکن است خراب شوند، منابع جدیدی ممکن است به ابرها بییوندند و یا راه‌اندازی مجدد شوند.

- **مجموعه زمانبندی وظیفه:** زمانبندی وظایف به دو بخش استراتژی زمانبندی و استراتژی بازگشت تقسیم می‌شود. ترکیبی از زمانبندی و استراتژی بازگشت منابع، مجموعه زمانبندی وظیفه را تشکیل می‌دهند.

۳. معیارهای ارزیابی زمانبندی در محیط‌های ابری

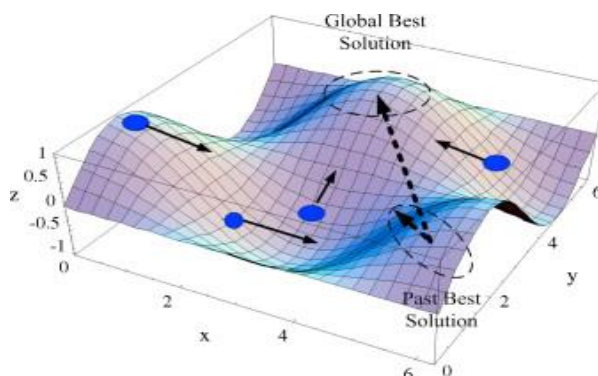
اهداف زمانبندی وظیفه در محاسبات ابری، ارائه زمانبندی بهینه برای کاربران و در همان زمان، ارائه توان عملیاتی سیستم ابر و کیفیت سرویس است [7]. اهداف خاص زمانبندی شامل: توازن بار، کیفیت خدمات، اصول اقتصادی، بهترین زمان اجرا و توان عملیاتی سیستم می‌باشند.

- **کیفیت خدمات:** ابر عمدتاً به کاربران، محاسبات و سرویس‌های ذخیره‌سازی ابر را ارائه می‌دهد، کاربران منابع را تقاضا و منابع توسط ارائه‌دهنده در قالب کیفیت خدمات عرضه می‌شوند. هنگامی که مدیریت زمانبندی وظیفه به سمت تخصیص وظیفه می‌رود، تضمین کیفیت سرویس منابع ضروری است [2].
- **اصول اقتصادی:** منابع محاسبات ابری به طور گسترده‌ای در سراسر جهان توزیع شده‌اند. این منابع ممکن است به سازمان‌های مختلف تعلق داشته باشند. آنها سیاست‌های مدیریتی خودشان را دارند. به عنوان یک مدل کسب‌وکار، محاسبات ابری، با توجه به شرایط مختلف، خدمات مربوطه را ارائه می‌دهد؛ بنابراین تقاضای عوارض معقول است. اقتصاد بازار زمانبندی وظایف و مدیریت منابع هدایت می‌کند، ما باید مطمئن شویم که نفع هر دو (مصرف‌کننده و ارائه‌دهنده) باشد، به طوری که محاسبات ابری بتواند بیشتر و بیشتر به جلو حرکت کند [8].
- **بهترین زمان اجرا:** در درجه اول برای برنامه‌های کاربردی، وظایف را می‌توان با توجه به نیازهای کاربران، به دسته‌های مختلف تقسیم و پس از آن بهترین زمان اجرا را، بر اساس اهداف مختلف برای هر وظیفه تنظیم کرد. آن کیفیت سرویس زمانبندی وظایف را به طور غیرمستقیم در محیط ابر بهبود خواهد بخشید [9].
- **توان عملیاتی سیستم:** عمدتاً برای سیستم‌های محاسبات ابری، توان عملیاتی، اندازه‌گیری عملکرد بهینه زمانبندی وظیفه سیستم است و آن نیز یک هدف است که باید در توسعه مدل کسب‌وکار در نظر گرفته شود. افزایش توان عملیاتی برای کاربران و ارائه‌دهندگان ابر سودی برای هر دو آنها خواهد داشت [10].

۴. الگوریتم بهینه‌سازی ازدحام ذرات

الگوریتم Particle Swarm Optimization، برای اولین بار توسط کندی و ابرهات در سال ۱۹۹۵ مطرح شد. PSO، یک الگوریتم محاسبه‌ای تکاملی الهام گرفته از طبیعت و براساس تکرار می‌باشد. منبع الهام این الگوریتم، رفتار اجتماعی حیوانات، همانند حرکت دسته جمعی پرندگان و ماهی‌ها بود. از این جهت که PSO نیز با یک ماتریس جمعیت تصادفی اولیه، شروع می‌شود، شبیه بسیاری دیگر از الگوریتم‌های تکاملی همچون الگوریتم ژنتیک پیوسته و الگوریتم رقابت استعماری است. برخلاف الگوریتم ژنتیک، PSO هیچ عملگر تکاملی همانند جهش و تقاطع ندارد. از این جهت می‌شود گفت که

الگوریتم رقابت استعماری شباهت بیشتری به PSO دارد تا به GA^* . هر عنصر جمعیت، یک ذره نامیده می‌شود (که همان معادل کروموزوم در GA و یا کشور در الگوریتم رقابت استعماری) است. در واقع الگوریتم PSO از تعداد مشخصی از ذرات تشکیل می‌شود که به طور تصادفی، مقدار اولیه می‌گیرند. برای هر ذره دو مقدار وضعیت و سرعت، تعریف می‌شود که به ترتیب با یک بردار مکان و یک بردار سرعت، مدل می‌شوند. این ذرات، بصورت تکرار شونده‌ای در فضای n بعدی مسئله حرکت می‌کنند تا با محاسبه مقدار بهینگی به عنوان یک ملاک سنجش، گزینه‌های ممکن جدید را جستجو کنند. شکل (۱) حرکت ذرات در الگوریتم PSO را نشان می‌دهد.



شکل (۱): حرکت ذرات در الگوریتم PSO

بعد فضای مسئله، برابر تعداد پارامترهای موجود در تابع مورد نظر برای بهینه‌سازی می‌باشد. یک حافظه به ذخیره بهترین موقعیت هر ذره در گذشته و یک حافظه به ذخیره بهترین موقعیت پیش آمده در میان همه ذرات، اختصاص می‌یابد. با تجربه حاصل از این حافظه‌ها، ذرات تصمیم می‌گیرند که در نوبت بعدی، چگونه حرکت کنند. در هر بار تکرار، همه ذرات در فضای n بعدی مسئله حرکت می‌کنند تا بالاخره نقطه بهینه سراسری پیدا شود. به‌روز رسانی سرعت، تحت تأثیر هر دو مقدار بهترین جواب محلی و بهترین جواب مطلق قرار می‌گیرد. بهترین جواب محلی و بهترین جواب مطلق، بهترین جواب‌هایی هستند که تا لحظه‌ی جاری اجرای الگوریتم، به ترتیب توسط یک ذره و در کل جمعیت به دست آمده‌اند. مزیت اصلی PSO این است که پیاده‌سازی این الگوریتم ساده بوده و نیاز به تعیین پارامترهای کمی دارد. همچنین PSO قادر به بهینه‌سازی توابع هزینه‌ی پیچیده با تعداد زیاد مینیمم محلی است.

۵. الگوریتم بهینه‌سازی تکامل تفاضلی

الگوریتم تکامل تفاضلی (DE) نخستین بار در سال ۱۹۹۵ توسط استورن و پرایس [11] معرفی شد. این دو نشان دادند که این الگوریتم توانایی خوبی در بهینه‌سازی توابع غیرخطی مشتق ناپذیر دارد که به عنوان روشی قدرتمند، ساده و سریع برای مسائل بهینه‌سازی در فضاهای پیوسته معرفی شده است. این روش تنها با تنظیم سه پارامتر شروع به کار می‌کند. یکی از روش‌های محاسبه توابع حقیقی (*Real value*) با استفاده از استراتژی‌های تکاملی است. روند تکامل در این الگوریتم

* Genetic Algorithm

مبتنی بر ایجاد بهبود تدریجی و مستمر در حدس اولیه (پاسخ کاندید) بوده و طبق اصول تمامی الگوریتم‌های رده تکاملی، به یک تابع برازندگی* جهت مقایسه پاسخ‌ها نیاز داریم.

نقطه قوت الگوریتم DE در مقایسه با روش‌های حل معادلات حقیقی دیگر (مانند روش‌های نیوتن)، عدم نیاز آن به گرادیان یا شیب تابع است. در نتیجه با استفاده از این الگوریتم، بدون وجود هرگونه اطلاعاتی در مورد نوع تابع می‌توان به محاسبه یک پاسخ نسبتاً بهینه برای انواع توابع چند بعدی پیوسته/غیر پیوسته، متغیر زمانی و نامنظم امیدوار بود.

این الگوریتم شباهت زیادی به Simulated Annealing دارد، با این تفاوت که صرفاً برای محاسبه توابع حقیقی طراحی شده است. الگوریتم DE جهت غلبه بر عیب اصلی الگوریتم ژنتیک، یعنی فقدان جستجوی محلی در این الگوریتم ارائه شده است، تفاوت اصلی بین الگوریتم‌های ژنتیک و الگوریتم (DE) در عملگر انتخاب[†] می‌باشد. در اپراتور انتخاب GA، شانس انتخاب یک جواب به عنوان یکی از والدین وابسته به مقدار شایستگی آن می‌باشد، اما در الگوریتم DE همه جواب‌ها دارای شانس مساوی جهت انتخاب شدن می‌باشند. یعنی شانس انتخاب شدن آن‌ها وابسته به مقدار شایستگی آن‌ها نمی‌باشد، پس از این که یک جواب جدید با استفاده از یک اپراتور جهش خودتنظیم و اپراتور crossover تولید شد، جواب جدید با مقدار قبلی مقایسه می‌شود و در صورت بهتر بودن جایگزین می‌گردد.

۶- روش پیشنهادی

الگوریتم ازدحام ذرات یک الگوریتم کارآمد در مسایل بهینه سازی می‌باشد که سرعت همگرایی بالایی دارد. این الگوریتم برای مسایلی که بهینه محلی زیادی ندارد یک الگوریتم مناسب بشمار می‌رود. چراکه در این قبیل مسایل ازدقت بالاتری نسبت به الگوریتم‌های مشابه دارد. اما در کنار مزایایی که عنوان شد. در این تحقیق از یک زمان بندی سه هدفه برای بهبود توالی وظایف بهره می‌گیریم. اهداف در نظر گرفته شده شامل زمان اجرای وظایف، تعادل بار و انرژی مصرفی ماشین‌های فیزیکی می‌باشد. الگوریتم ازدحام ذرات در کنار مزایای و ویژگی‌هایی که دارد با چالش گرفتار شدن در بهینه‌های محلی و کاهش تنوع جمعیت مواجه است که بر کارایی آن اثر منفی می‌گذارد [12] بر همین اساس در روش پیشنهادی، سعی خواهد شد؛ با بهره‌گیری از قابلیت‌های دیگر الگوریتم‌های بهینه سازی بر محدودیت‌های الگوریتم بهینه سازی ازدحام ذرات فائق آمده و رویکردی جدید برای زمانبندی وظایف در محیط‌های ابری پیشنهاد گردد. در روش پیشنهادی محدودیت گرفتار شدن در بهینه‌های محلی با استفاده از عملگرهای الگوریتم بهینه سازی تکامل تفصالی حل خواهد شد. روش پیشنهادی از چندگام گام‌های تشکیل شده است که عبارتند از:

۱.۶. مرحله اول: ساختارنمایش ذرات

ساختارنمایش ذره به منظور کد کردن مسئله از دنیای واقعی به ساختاری جهت آنالیز می‌باشد. در این تحقیق هر ذره یک بردار به طول n است که هر بعد آن بیان‌گر گره فیزیکی می‌باشد که روی وظیفه جاری اجرا می‌شود. بنابراین هر وظیفه در ساختار ذره پیشنهادی به یک گره فیزیکی نگاشت می‌شود و مدیریت منبع در گره فیزیکی، وظیفه تخصیص منابع موجود در ماشین‌های مجازی را به وظایف نگاشت شده برعهده دارد.

* Fitness function

† selection operator

ساختار نمایش ذره به منظور کد کردن مسئله از دنیای واقعی به ساختاری جهت آنالیز می‌باشد. در این تحقیق هر ذره یک بردار به طول n است که هر بعد آن بیان‌گر گره فیزیکی می‌باشد که روی وظیفه جاری اجرا می‌شود. بنابراین هر وظیفه در ساختار ذره پیشنهادی به یک گره فیزیکی نگاشت می‌شود و مدیریت منبع در گره فیزیکی، وظیفه تخصیص منابع موجود در ماشین‌های مجازی را به وظایف نگاشت شده بر عهده دارد. در واقع هر راه حل معادل یک بردار است که درایه X_i برابر ماشین مجازی انجام دهنده وظیفه i می‌باشد. شکل (۲) یک راه‌حل را با ۴ ماشین مجازی و ۸ وظیفه نشان می‌دهد.

W1	W2	W3	W4	W5	W6	W7	W8
۲	۱	۳	۲	۴	۱	۲	۳

شکل (۲): یک راه حل در روش پیشنهادی

در این مدل اندیس هر عنصر از بردار، بیانگر شماره وظیفه و مقدار عناصر بردار، بیانگر ماشین مجازی تخصیص داده شده به هر وظیفه می‌باشد. پس هر راه حل برداری می‌باشد که طول آن برابر با تعداد وظایف می‌باشد.

۲.۶. مرحله دوم: ایجاد جمعیت اولیه بالگوریتم بهینه سازی ازدحام ذرات

الگوریتم یکی از PSO کار خود را با تولید جمعیت اولیه* از ذرات آغاز می‌کند. الگوریتم‌های PSO جمعیت اولیه را تصادفی ایجاد می‌کند که در این صورت ذرات کیفیت پایینی دارند. به منظور رفع این مشکل در این تحقیق از یک الگوریتم حریصانه[†] استفاده می‌شود. این الگوریتم براساس وضعیت فعلی سیستم زمان‌بند بهینه‌ترین گره فیزیکی را با توجه به اولویت هر وظیفه انتخاب می‌کند.

۳.۶. مرحله سوم: محاسبه برازندگی راه حل ها

باتوجه به این نکته که تعادل بار مستلزم فراهم کردن منابع مورد نیاز کاربران می‌باشد، زمان‌بندی متعادل با کاهش زمان اتمام کل کارها میسر خواهد بود. برازندگی راه‌حل‌ها که با یک تابع ارزیابی[‡] انجام می‌شود به منظور رمزگشایی ذرات و تعیین میزان بهینگی آنها با توجه به توالی وظایف می‌باشد. در واقع سیستم زمان‌بندی وظایف در محیط ابر یک مسئله بهینه‌سازی چند هدفه است که ما برای بررسی یک توالی خاص از وظایف، چهار هدف «حداقل‌سازی زمان اجرا وظایف»، «حداقل‌سازی زمان پاسخ»، «حداقل‌سازی زمان اتمام وظایف (MakeSpan)» و «تعادل بار با کاهش نرخ مهاجرت» را در نظر می‌گیریم. با این شرایط راه‌حل‌هایی با کمترین زمان پاسخ، کمترین واریانس بار بین گره‌ها فیزیکی، کمترین زمان اجرا و کمترین نرخ مهاجرت باشند، بهترین مقدار برازندگی به آنها تخصیص داده می‌شود.

۴.۶. مرحله چهارم: بروزرسانی معادله موقعیت و سرعت ذره

برای به روز رسانی موقعیت و سرعت هر کدام از ذرات از رابطه ۱ و ۲ استفاده می‌شود.

(۱)

$$V_i(t+1) = w * V_i(t) + c_1 * r_1 * (P_{i.best} - X_i(t)) + c_2 * r_2 * (P_{g.best} - X_i(t))$$

(۲)

$$X_i = X_i(t-1) + V_i(t)$$

* Population initialization

† Greedy algorithm

‡ Evaluation Function

W ضریب وزنی اینرسی (حرکت در مسیر خودی) که نشان دهنده میزان تأثیر بردار سرعت تکرار قبل $(V_i(t))$ بر روی بردار سرعت در تکرار فعلی $(V_i(t+1))$ است. C_1 و C_2 ضریب ثابت آموزش (حرکت در مسیر بهترین تجربه شخصی و بهترین ذره یافت شده در بین کل جمعیت) و r_1 و r_2 دو عدد تصادفی با توزیع یکنواخت در بازه $(0, 1)$ می باشد.

۵.۶. مرحله پنجم بروزرسانی بهترین تجربه شخصی و بهترین تجربه سراسری

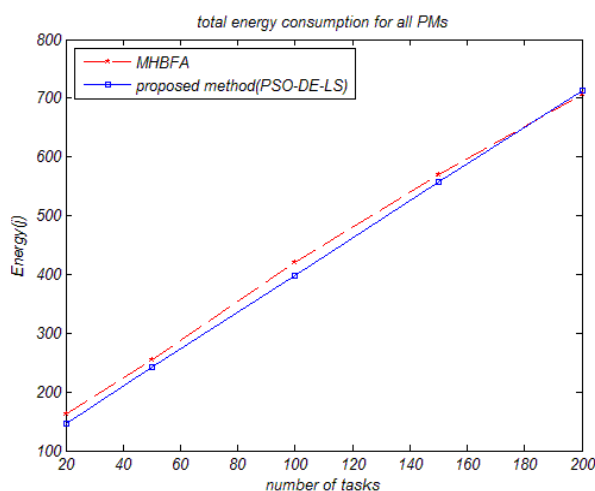
بروزرسانی بهترین تجربه شخصی (Pbest) با استفاده از عملگر برش و جهش در الگوریتم DE بهترین تجربه شخصی با استفاده فرمول جهش و برش زیر بروزرسانی می شوند. در مرحله هفتم جمعیت نسل بعدی شرط خاتمه مدنظر می باشد.

۷. ارزیابی و مقایسه

رایانش ابری شامل مجموعه ای از مراکز داده ای است و هر مرکز داده شامل مجموعه n میزبان است و هر میزبان شامل مجموعه ای از m ماشین مجازی می باشد. هر مرکز داده شامل تعدیل کننده بار VM می باشد که مسئول یافتن میزبان و ماشین مجازی مناسب جهت تخصیص وظیفه بعدی از طریق یافتن برخی معیارها می باشد. مهمترین معیارهای استفاده شده در این مقاله زمان اجرای کل، شاخص عدم تعادل و انرژی مصرفی است. ارزیابی کارایی روش پیشنهادی در زمانبندی وظایف محیط های ابری براساس تاثیر تعداد وظایف صورت پذیرفته است. متغیر مورد نظر، تعداد مختلف وظایف می باشد؛ تا تاثیر مقادیر مختلف وظایف بر کارایی روش پیشنهادی بررسی گردد. برای این منظور تعداد وظایف در محیط ابری ۲۰، ۴۰، ۶۰، ۸۰، ۱۰۰، ۱۲۰، ۱۴۰، ۱۶۰، ۱۸۰ و ۲۰۰ در نظر گرفته شده و آزمایش در شرایط یکسان برای روش پیشنهادی و روش پایه در مرجع [13] انجام گرفت. نتایج این بررسی به ازای هر کدام از معیارهای ارزیابی به صورت مستقل ذکر گردیده است.

۱.۷. انرژی مصرفی برای کل ماشین های مجازی

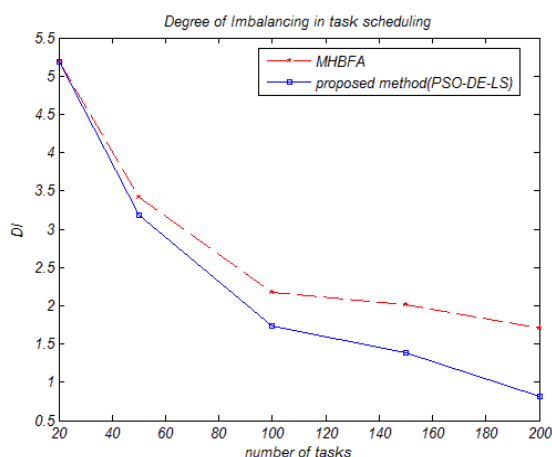
نتایج نشان دهنده کارایی مناسب زمانبندی وظایف مبتنی بر الگوریتم بهینه سازی ترکیبی ازدحام ذرات و تکامل تفاضلی می باشد. ارزیابی به ازای تعداد مختلفی از وظایف (۲۰، ۴۰، ۶۰، ۸۰، ۱۰۰، ۱۲۰، ۱۴۰، ۱۶۰، ۱۸۰ و ۲۰۰) انجام شده است. یافته ها نشان می دهد روش پیشنهادی به صورت میانگین (به ازای تعداد مختلفی از ماشین های مجازی) ۹ درصد زمان کمتری نسبت به روش پایه برای اجرای کارها نیاز دارد. شکل (۳) نتایج این بررسی را نشان می دهد. در این نمودار محور عمودی بیانگر زمان اجرای کارها به ازای هر کدام از روش ها و محور افقی بیانگر تعداد وظایف می باشد.



شکل (۳): بررسی انرژی مصرفی ماشین های فیزیکی مرکز داده ابری به ازای تعداد مختلف وظایف

۲.۷. ضریب عدم تعادل بار در زمانبندی وظایف

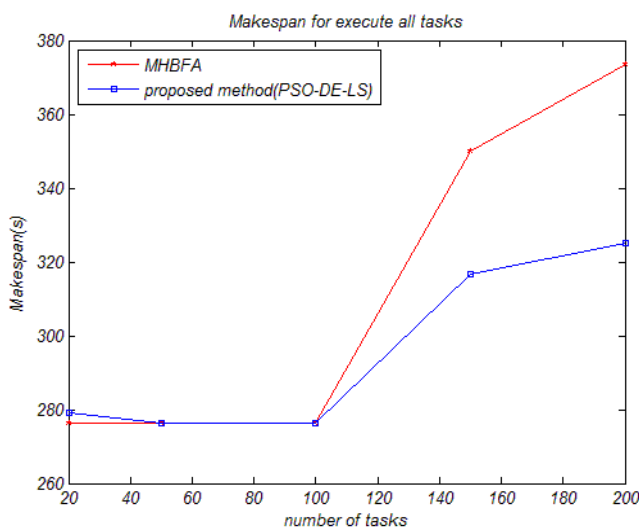
نتایج در معیار ارزیابی ضریب عدم تعادل بار در فرآیند زمانبندی وظایف نشان می‌دهد که روش زمانبندی وظایف مبتنی بر الگوریتم بهینه‌سازی ترکیبی ازدحام ذرات و تکامل تفاضلی نسبت به روش پایه تعادل بار بیشتری در محیط ابری ایجاد نماید. ارزیابی به ازای تعداد مختلفی از وظایف (۲۰، ۴۰، ۶۰، ۸۰، ۱۰۰، ۱۲۰، ۱۴۰، ۱۶۰، ۱۸۰ و ۲۰۰) انجام شده است. یافته‌ها نشان می‌دهد روش پیشنهادی به صورت میانگین ۱۶ درصد تعادل بار بهتری را نسبت به روش پایه در محیط ابری ایجاد نماید. دیگر بررسی نیز بیانگر آن است که در فرآیند زمانبندی وظایف تعادل بار محیط ابری در روش پیشنهادی همواره مساوی یا کمتر از روش پایه می‌باشد. شکل (۴) نتایج این بررسی را نشان می‌دهد. در این نمودار محور عمود بیانگر ضریب عدم تعادل بار به ازای هر کدام از روش‌ها و محور افقی بیانگر تعداد وظایف می‌باشد.



شکل (۴): بررسی ضریب عدم تعادل بار در زمانبندی وظایف برای روش پیشنهادی و پایه

۳.۷. زمان اجرای کل

ارزیابی زمان اجرای کل نشان می‌دهد که روش زمانبندی وظایف مبتنی بر الگوریتم بهینه‌سازی ترکیبی ازدحام ذرات و تکامل تفاضلی نسبت به روش پایه کارها/وظایف را در زمان کمتری اجرا نماید. در این سناریو آزمایش به ازای تعداد مختلفی از وظایف (۲۰، ۴۰، ۶۰، ۸۰، ۱۰۰، ۱۲۰، ۱۴۰، ۱۶۰، ۱۸۰ و ۲۰۰) انجام شده است. یافته‌ها نشان می‌دهد روش پیشنهادی به صورت میانگین ۵ درصد زمان کمتری نسبت به روش پایه برای اجرای کارها نیاز دارد. شکل (۵) نتایج این بررسی را نشان می‌دهد. در این نمودار محور عمود بیانگر زمان اجرای کارها به ازای هر کدام از روش‌ها و محور افقی بیانگر تعداد وظایف می‌باشد.



شکل (۵): بررسی زمان اجرای کل به ازای هر کدام از روش‌های پیشنهادی و پایه

۸. نتیجه گیری

رایانش ابری، مدلی کامپیوتری است که دسترسی کاربران را بر اساس نوع تقاضایی که از منابع اطلاعاتی و محاسباتی دارند محیا می‌کند. این مدل سعی دارد با کمترین نیاز به منابع و کاهش هزینه‌ها و افزایش سرعت دسترسی اطلاعات، جوابگوی نیاز کاربران باشد. مدیریت منابع و تخصیص وظایف در کنار امنیت، قابلیت اطمینان و حفظ اعتماد مشترکین یکی از چالش‌های مهم در زمینه رایانش ابری می‌باشد که می‌تواند روی سایر مسائل نیز تاثیرگذار باشد. واضح است که تعداد کارها و تعداد منابع در محیط ابر می‌تواند بسیار گسترده باشد و به همین علت ترتیب اجرای کارها و نحوه تخصیص منابع تاثیر مهمی بر کارایی سرور ابر دارد. هر لحظه در محیط بصورت همزمان تعدادی درخواست ارسال می‌گردد که برای پاسخ به هر یک از آنها باید یک یا چند کار داده شود و هر یک از این کارها به منابع خاصی نیاز دارند، که در این رابطه روش‌های مختلفی ارائه شده است. در این مقاله یک راهکار مبتنی بر الگوریتم‌های بهینه‌سازی برای زمانبندی وظایف در محیط‌های ابری پیشنهاد گردید. برای این منظور یک نسخه جدید از الگوریتم بهینه‌سازی ازدحام ذرات معرفی شد که در آن با بهره‌گیری از عملگرهای الگوریتم بهینه‌سازی تکامل تفاضلی سعی شد بر محدودیت‌های الگوریتم بهینه‌سازی ازدحام ذرات غلبه گردد. بررسی‌ها نشان داد که روش پیشنهادی در معیارهای ارزیابی کارایی مطلوبی دارد.

۹. مراجع

- [1] F. Ramezani, J. Lu, and F. K. Hussain, "Task-based system load balancing in cloud computing using particle swarm optimization," *International journal of parallel programming*, vol. 42, no. 5, pp. 739-754, 2014.
- [2] S. Singh and I. Chana, A survey on resource scheduling in cloud computing: Issues and challenges, *Journal of grid computing*, Vol. 14, No. 2, pp. 217-264, 2021.

- [3] N. J. Navimipour, Task scheduling in the cloud environments based on an artificial bee colony algorithm, In Proceedings of the International Conference on Image Processing, Production and Computer Science, Istanbul (Turkey) 2022.
- [4] M. Kalra and S. Singh, "A review of metaheuristic scheduling techniques in cloud computing," Egyptian informatics journal, vol. 16, no. 3, pp. 275-295, 2015.
- [5] A. Awad, N. El-Hefnawy, and H. Abdel_kader, "Enhanced particle swarm optimization for task scheduling in cloud computing environments," Procedia Computer Science, vol. 65, pp. 920-929, 2015.
- [6] J. Zhao, K. Yang, X. Wei, Y. Ding, L. Hu, and G. Xu, "A heuristic clustering-based task deployment approach for load balancing using Bayes theorem in cloud environment," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 305-316, 2016.
- [7] N. J. Navimipour and F. S. Milani, "Task scheduling in the cloud computing based on the cuckoo search algorithm," International Journal of Modeling and Optimization, vol. 5, no. 1, 2015.
- [8] Z.-H. Zhan, X.-F. Liu, Y.-J. Gong, J. Zhang, H. S.-H. Chung, and Y. Li, "Cloud computing resource scheduling and a survey of its evolutionary approaches," ACM Computing Surveys (CSUR), vol. 47, no. 4, 2015.
- [9] M. Masdari, S. ValiKardan, Z. Shahi, and S. I. Azar, "Towards workflow scheduling in cloud computing: a comprehensive analysis," Journal of Network and Computer Applications, vol. 66, pp. 64-82, 2016.
- [10] T. Wang, Z. Liu, Y. Chen, Y. Xu, and X. Dai, "Load balancing task scheduling based on genetic algorithm in cloud computing," in 2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing (DASC),, 2014, pp.146-152: IEEE.
- [11] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," Journal of global optimization, vol. 11, no. 4, pp. 341-359, 1997.
- [12] M. Xu, W. Tian, and R. Buyya, "A survey on load balancing algorithms for virtual machines placement in cloud computing," Concurrency and Computation: Practice and Experience, vol. 29 ,no. 12, 2021.
- [13] S. Srichandan, T. A. Kumar, S. Bibhudatta, Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm, Future Computing and Informatics Journal, Vol. 3, No. 2, pp. 210-230, 2018.