

زمان بندی با آگاهی از عدم قطعیت جریان کاری های بلادرنگ مبتنی بر خوشه بندی سطحی افقی تحت محدودیت مهلت در سیستم های چند ابری

مرضیه سادات افضلی

دانشجوی کارشناسی ارشد مهندسی فناوری اطلاعات گرایش تجارت الکترونیکی موسسه آموزش عالی اترک قوچان

دکتر مسعود خسروی فارمد

مدرس گروه مهندسی فناوری اطلاعات موسسه آموزش عالی اترک قوچان

چکیده

زمان بندی جریان کاری در سیستم های چند ابری نیز مورد توجه بسیاری از محققان قرار گرفته است. با این حال، به دلیل انواع منابع ناهمگن و مکانیسم های صدور صورت حساب چندگانه، چالش برانگیز است. در سال های اخیر، فناوری محاسبات ابری گسترش یافته است و به طور گسترده مورد استفاده قرار گرفته است. بنابراین یکی از مسائلی که در توسعه رایانش ابری باید حل شود، واگذاری وظایف به ماشین های مجازی مناسب و کاهش زمان انتظار بیکار ماشین های مجازی اجاره ای برای به دست آوردن هزینه اجاره بهینه است. در این مقاله الگوریتم زمان بندی با آگاهی از عدم قطعیت جریان کاری بلادرنگ مبتنی بر خوشه بندی سطحی افقی تحت محدودیت مهلت در سیستم های چند ابری ارائه شده است. روش پیشنهادی از یک روش خوشه بندی افقی و عمودی استفاده می کند. خوشه بندی افقی هزینه را کاهش می دهد و خوشه بندی عمودی زمان اجرا را کاهش می دهد. هدف الگوریتم پیشنهادی به حداقل رساندن هزینه اجاره اجرای وظایف با رعایت محدودیت مهلت زمانی جریان کاری است. بطوریکه درصد نقض مهلت کاهش و کارایی استفاده ماشین مجازی را افزایش می دهد. شبیه سازی در محیط متلب برای روش پیشنهادی و روش پایه برای متغیرهای هزینه، زمان اجرا درصد نقض مهلت و کارایی استفاده ماشین مجازی انجام شد. هر دو الگوریتم ارزیابی شدند و نتایج نشان داد که روش پیشنهادی عملکرد بهتری نسبت به روش پایه دارد.

واژگان کلیدی: زمان بندی چند ابری، عدم قطعیت، خوشه بندی سطحی افقی، بلادرنگ

۱. مقدمه

قابلیت انعطاف پذیری و ویژگی های رایانش ابری در بین مشتریان محبوب است و برنامه های جریان کاری بیشتری به پلتفرم های ابری مهاجرت می کنند. هدف بسیاری از الگوریتم های زمان بندی جریان کاری دستیابی به حداقل هزینه های اجاره است. با این حال، اکثر تحقیقات موجود فرض می کنند که زمان اجرای کار قطعی است. در واقع، به دلیل نوسانات عملکرد ماشین های مجازی، زمان اجرای وظیفه قبل از زمان بندی نامشخص است. علاوه بر این، بسیاری از کارها صرفه جویی در هزینه سیستم های چند ابری را نادیده می گیرند (Xu et al, 2023).

زمان بندی جریان کاری در سیستم های چند ابری نیز مورد توجه بسیاری از محققان قرار گرفته است. با این حال، به دلیل انواع منابع ناهمگن و مکانیسم های صدور صورت حساب چندگانه، چالش برانگیز است (Armbrust et al, 2010). در سال های اخیر، فناوری محاسبات ابری گسترش یافته است و به طور گسترده مورد استفاده قرار می گیرد. این فناوری دارای ویژگی های مجازی سازی، انعطاف پذیری سریع و پرداخت هزینه است که باعث می شود مشتریان ترجیح دهند از خدمات ارائه شده توسط ارائه دهندگان ابری به جای خرید و نگهداری سرورهای مربوطه استفاده کنند. در نتیجه استفاده از زیرساخت ارائه دهنده ابر افزایش می یابد و هزینه های مشتری نیز کاهش می یابد (Masdari et al, 2016).

وظایفی که در یک ماشین مجازی اجرا می شوند ممکن است به دلایل مختلف از جمله سخت افزار، عدم تطابق منابع و سایر خرابی ها با شکست مواجه شوند (Dogan and Ozguner, 2002). برای اطمینان از قابلیت اطمینان اجرای کار، می توان از تکنیک های تکثیر یا نقطه بازرسی استفاده کرد، که در آن اولی با ایجاد پشتیبان گیری از کار، احتمال شکست را کاهش می دهد، و دومی نقاط بازرسی را در حین اجرای کار وارد می کند و زمانی که کار با شکست مواجه می شود، اجرا را از نقطه بازرسی قبلی از سر می گیرد (Hadeer et al, 2020). با این حال، این هزینه های اضافی را در رابطه با هزینه کل اجاره و زمان اجرای کار متحمل می شود. بنابراین، باید در طراحی الگوریتم زمان بندی این سرورها را در نظر بگیریم (Xie et al, 2017).

با رشد سرویس های مدیریت توزیع محتوا و محاسبات تعاملی، مانند شبکه های اجتماعی و فرآیندهای علمی آنلاین، ظرفیت مراکز داده ابری محدود شده و نمی تواند پاسخگوی نیازهای تجاری در ساعات اوج مصرف باشد. برای مدیریت این حجم عظیم داده، سیستم های ابری متعددی معرفی شده اند که چندین ابر را با هم ترکیب می کنند تا یک مجموعه واحد از خدمات را به روشی مشترک ارائه دهند (Chen, 2023).

مساله اصلی تحقیق اجرای جریان کاری در محدودیت مهلت زمانی با بهبود معیارهای کیفیت سرویس است. همچنین هزینه اجاره منبع، زمان اجرای جریان کاری، نرخ نقض مهلت زمانی و کارایی استفاده ماشین مجازی جزء مسائل تحقیق می باشد. در راستای این محدودیت ها یک الگوریتم زمان بندی با آگاهی از عدم قطعیت جریان کاری بلادرنگ تحت محدودیت های مهلت در سیستم های چند ابری ارائه شد. هدف این الگوریتم کاهش هزینه اجاره ماشین های مجازی برای اجرای جریان کاری در مهلت زمانی معین است و در ادامه بهبود نرخ نقض مهلت و کارایی استفاده از منابع ماشین مجازی است. این مقاله محققان را قادر می سازد تا رویکرد مناسبی را برای مطالعه و تحقیق انتخاب نمایند و پیشنهاد طرح های بهتری برای زمان بندی برنامه کاربر ارائه دهند. موضوعات تحقیقاتی آینده نیز در این پژوهش پیشنهاد شده است.

در ادامه این مقاله، بخش ۲ به مرور ادبیات مدل های زمان بندی جریان کاری در رایانش ابری و کارهای مرتبط می پردازد. بخش ۳ به روش پیشنهادی می پردازد و بخش ۴ شامل مقایسه و تحلیل نتایج الگوریتم پیشنهادی و الگوریتم پایه است. در نهایت، نتیجه گیری و پیشنهاد کارهای آتی در بخش ۵ ارائه می شود.

۲. کار مرتبط

محققان به طور گسترده مساله زمان بندی جریان کاری را مطالعه کرده اند (Abrishami et al, 2013)، و دستیابی به یک راه حل زمان بندی بهینه به دلیل وابستگی های بین وظایف است (WuQ et al, 2017, Juve et al, 2013). بیشتر تحقیقات موجود بر روی

ZhangR and ShiW, 2021, Alkhanak et al) در زمان اجرای کلی جریان کار تمرکز دارند (al, 2016).

اکثر تحقیقات بر زمان بندی یک جریان کاری تحت محدودیت های کیفیت سرویس تعریف شده توسط کاربران، تمرکز می کنند و فرض می کنند که زمان اجرای کار در یک ماشین مجازی قطعی است (Cai et al, 2017).

هنگام رسیدگی به مساله زمان بندی جریان کاری بلادرنگ، به این معنی که جریان کاری متعددی برنامه ریزی می شوند و هر کدام به طور تصادفی وارد می شوند (Ding et al, 2018). این الگوریتم ها زمان بی کاری را در ماشین های مجازی اجاره ای هدر می دهند، زیرا چندین کار جریان کاری را برای یک ماشین مجازی زمان بندی نمی کنند (Arabnejad et al, 2019). علاوه بر این، زمان اجرای وظایف به دلیل نوسان عملکرد ماشین های مجازی در محیط های ابری نامشخص است (Ma et al, 2021, Liu et al, 2021).

در پژوهش (Xu et al, 2023) زمان بندی با آگاهی از عدم قطعیت جریان کاری بلادرنگ تحت محدودیت های مهلت در سیستم های چند ابری را ارائه شد. چارچوب الگوریتم پیشنهاد شده شامل چهار جزء اصلی است: تجزیه و تحلیل جریان کاری، مجموعه وظایف، کنترل کننده تخصیص وظایف، و مدیر منابع. سپس بر اساس این چارچوب، الگوریتم اکتشافی RWSMC¹ پیشنهاد می شود. هدف این الگوریتم به حداقل رساندن کل هزینه اجاره و در عین حال رعایت محدودیت های مهلت و اطمینان از قابلیت اطمینان اجرای کار است. الگوریتم RWSMC با انتخاب مکانیسم صورت حساب مناسب بر اساس زمان اجرای کار، هزینه را کاهش می دهد و با زمان بندی کار در ماشین مجازی با کوتاه ترین زمان شروع پیش بینی شده، تأثیر زمان اجرا را کاهش می دهد.

در پژوهش (Taghinezhad et al, 2022) زمان بندی جریان کاری با انرژی کارآمد با محدودیت های مهلت بودجه برای ابر را ارائه شد. در این مقاله، دو الگوریتم اکتشافی کارآمد انرژی با محدودیت های مهلت بودجه که برای منابع با ولتاژ پویا و مقیاس بندی فرکانس فعال DVFS² ارائه شده است. الگوریتم های انرژی آگاه با محدودیت مهلت و بودجه BDCE³ و BDD⁴ پیشنهاد شده است.

در پژوهش (Chakravarthi et al, 2022) استراتژی زمان بندی جریان کاری آگاه از بودجه قابل اعتماد در محیط چند ابری ارائه شده است. نویسندگان این مقاله یک الگوریتم زمان بندی جریان کاری با محدودیت بودجه و قابلیت اعتماد NRBWS⁵ مبتنی بر نرمال سازی را برای بهبود قابلیت اطمینان اجرای جریان کاری و کاهش طول عمر تحت محدودیت بودجه مشخص شده توسط کاربر را پیشنهاد می کنند. این طرح تحت یک فرآیند نرمال سازی حداقل تا حداکثر است که توسط محاسبه بودجه معقول مورد انتظار برای اختصاص وظایف به یکی از منابع محاسباتی دنبال می شود.

در پژوهش (Deldari et al, 2022) الگوریتم زمان بندی چند هسته ای جدید برای جریان کاری علمی با محدودیت مهلت آگاه از هزینه در ابر زیرساخت به عنوان سرویس ارائه شده است. نویسندگان این مقاله یک الگوریتم زمان بندی اکتشافی، تقسیم خوشه ای CDA⁶ را پیشنهاد می کند که بر گسترش استفاده از منابع چند هسته ای برای کاهش هزینه های اجرا و در عین حال رعایت مهلت تعیین شده توسط کاربر تمرکز دارد.

در پژوهش (Xiao Ge and Qiang Zhang, 2022) یک الگوریتم بارگذاری محاسباتی بهینه با محدودیت مصرف انرژی برای سیستم چند ابری ارائه شده است. استراتژی پیشنهادی ابتدا وظایف را مرتب می کند و مسیرها را تعیین می کند. سپس وظایف در هر مسیر تعیین شده به عنوان یک وظیفه یکپارچه شمارش می شود و به صورت تکراری به یک گره محاسباتی با زمان تکمیل کمتر اختصاص داده می شود.

در پژوهش (Chen, 2023) روش زمان بندی وظایف بهینه سازی چند هدفه بر اساس برنامه نویسی پویا برای محیط چند ابری را ارائه شده است. این روش ابتدا فرآیند تخصیص منابع را در یک محیط چند ابری توصیف می کند، سپس یک مدل پیشرفت برای وظایف چند

¹ Real-time workflows on Multi-Cloud Systems

² Dynamic Voltage and Frequency Scaling

³ Budget Deadline Constrained Energy aware

⁴ Budget Deadline DVFS-enabled energy-aware

⁵ Normalization based Reliable Budget constraint Workflow Scheduling

⁶ Cluster Dividing Algorithm

کاربره در یک محیط ابری ایجاد می‌کند، سپس مسئله بهینه‌سازی کار چند کاربر را رسمیت می‌دهد و یک الگوریتم راه‌حل را بر اساس رویکرد برنامه‌نویسی پویا طراحی می‌کند. نتایج تجربی نشان می‌دهد که این روش می‌تواند به بهینه‌سازی زمان و هزینه در زمان‌بندی وظایف چند کاربره دست یابد.

در پژوهش (Tang, 2022) استراتژی زمان‌بندی جریان کاری علمی مقرون‌به‌صرفه با قابلیت اطمینان در سیستم‌های چند ابری ارائه شده است. ابتدا یک چارچوب زمان‌بندی جریان کاری مقاوم در برابر خطای سیستم‌های چند ابری ساخته می‌شود، که تلاش می‌کند قابلیت اطمینان اجرای برنامه‌های علمی را بهبود بخشد و هزینه اجرای آن‌ها را کاهش دهد.

در پژوهش (Zhang et al, 2023) یک روش زمان‌بندی برای وظایف و سرویس‌ها در محیط‌های صنعتی اینترنت اشیاء چند ابری ارائه شده است. الگوریتمی به نام الگوریتم ژنتیک برنامه‌ریزی چند ابری ¹MCASGA پیشنهاد می‌کند که می‌تواند هم سرویس‌ها و هم وظایف را همزمان زمان‌بندی کند.

در پژوهش (Manam et al, 2023) به حداقل رساندن هزینه با مهلت محدود برای محیط‌های رایانش ابری ارائه شده است. در این مقاله، یک الگوریتم حداقل‌سازی هزینه با محدودیت زمانی ²DCCM برای زمان‌بندی منابع در رایانش ابری پیشنهاد شده است. در طرح پیشنهادی، وظایف براساس محدودیت‌های مهلت زمان‌بندی و وابستگی داده‌ها گروه‌بندی شدند. هدف اصلی الگوریتم پیشنهادی به حداقل رساندن هزینه در حین رعایت محدودیت‌های بودجه و مهلت است.

در پژوهش (Pradhan et al, 2022) الگوریتم زمان‌بندی وظایف ابری آگاه از انرژی در محیط چند ابری ناهمگن را ارائه شده است. در این مقاله، مسئله مصرف انرژی دیتاسنترهای ابری مورد بررسی قرار گرفته است و یک الگوریتم زمان‌بندی وظایف ابری آگاه از انرژی EACTS³ پیشنهاد شده است.

در پژوهش (Farid et al, 2023) زمان‌بندی جریان کاری علمی در چند ابری با رویکرد تصمیم‌گیری بهینه‌سازی حداقل وزن چند هدفه ارائه دادند. در این تحقیق، روش پارتو با استفاده از رویکرد حداقل وزن چند هدفه جدید محاسبه شده است. از بهینه‌سازی ازدحام ذرات برای گسترش الگوریتم زمان‌بندی چندهدفه⁴ FR-MOS با استفاده از مدیریت منابع فازی برای به حداکثر رساندن تنوع و به دست آوردن هم‌گرایی پارتو بهینه استفاده می‌شود.

۳. روش پیشنهادی

در این بخش ابتدا مدل پیشنهادی بیان می‌شود سپس روش پیشنهادی ارائه و جزئیات آن ارائه می‌شود.

۱.۳. مدل پیشنهادی

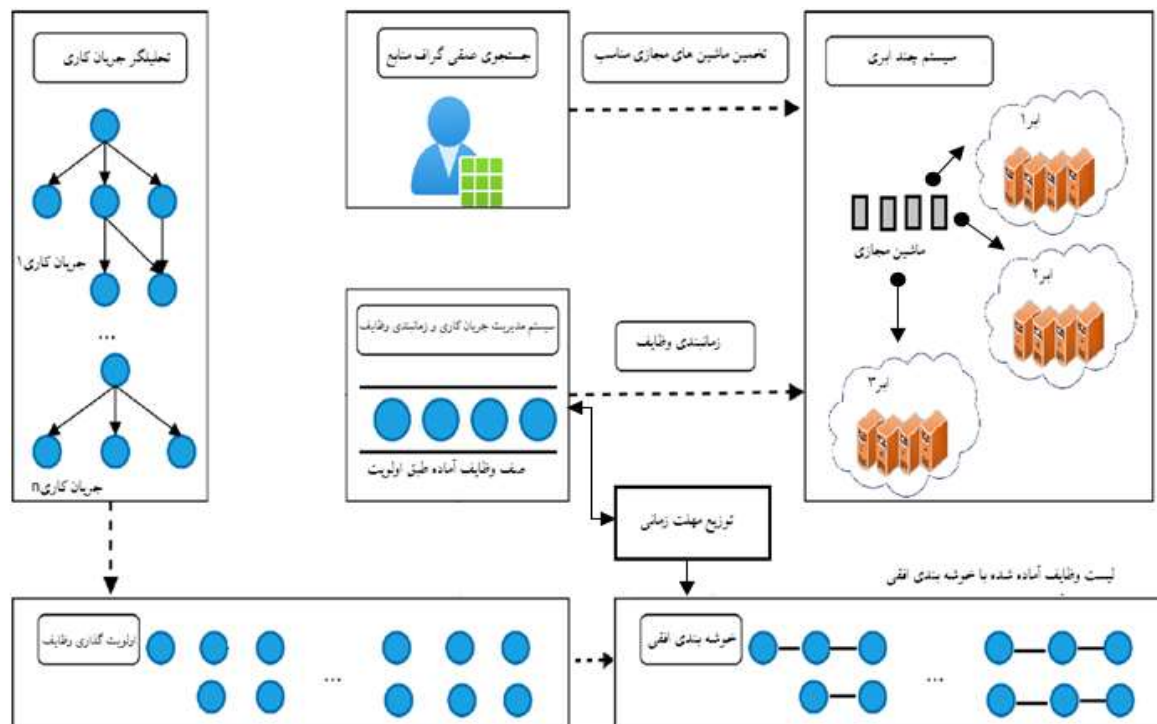
در این بخش از تحقیق به ارائه مدل پیشنهادی می‌پردازیم. شکل ۱ مدل پیشنهادی تحقیق را نشان می‌دهد. جریان کاری‌های مختلف به ابر ارسال می‌شوند. جریان کاری‌های دریافتی در بخش تحلیلگر جریان کاری جهت دسته‌بندی مورد تحلیل و ارزیابی قرار می‌گیرند. در بخش بعدی وظایف جریان کاری اولویت‌گذاری می‌شوند. سپس جریان کاری‌ها بصورت افقی خوشه‌بندی می‌شوند. مهلت زمانی بین سطح‌ها و وظایف توزیع می‌شوند. در بخش سیستم مدیریت جریان کاری تمام وظایف آماده طبق اولویت آن‌ها برای زمان‌بندی انتخاب می‌شوند. در بخش جستجوی عمقی گراف هر وظیفه از صف اولویت آماده انتخاب و بهینه‌ترین منبع تخصیص می‌یابد. در سیستم چند ابری منابع در قالب ماشین مجازی به کاربران ارائه می‌شود.

1 Multi-Cloud Application Scheduling Genetic Algorithm

2 Deadline-Constrained Cost Minimisation

3 Energy-Aware Cloud Task Scheduling

4 fuzzy resource multi-objective scheduling



۱: مدل پیشنهادی تحقیق

۲.۳. مدل منابع

مدل منابع شامل m نوع منبع ناهمگون $R = \{r_1, r_2, \dots, r_m\}$ با توان محاسباتی و حافظه مختلف ارائه می شود که هزینه هر منبع برای یک ساعت به صورت $Cost = \{cost_{r1}, cost_{r2}, \dots, cost_{rm}\}$ می باشد. یک منبع ناهمگون دارای مجموعه ای از پردازنده های موجود است و هر یک از آنها یک قیمت واحد در هر زمان دارد. قیمت پردازنده به گونه ای تعریف شده است که قدرتمندترین پردازنده دارای بالاترین هزینه است و پردازنده با توان پردازشی کم، کمترین هزینه را دارد.

۳.۳. مدل جریان کاری

مدل جریان کاری یکی از موفق ترین پارادایم ها برای برنامه نویسی برنامه های علمی در زیرساخت های توزیع شده ابری است. از آنجایی که وظایف گراف جریان کاری بردار یک طرفه متصل هستند، شکل کاربردی مورد استفاده برای یک جریان کاری، یک گراف غیر چرخه ای جهت دار $G = (T, E)$ است، که در آن رئوس گراف $T = \{t_0, t_1, t_2, \dots, t_n\}$ به آن اشاره دارد. $E = \{e_{ij} | t_i, t_j \in T\}$ یال های بین رئوس را نشان می دهد. هر راس دارای وزنی است که نشان دهنده حجم دستورالعمل های اجرایی است. همچنین هر یال دارای وزنی است که اولویت و میزان انتقال داده بین وظایف t_i و t_j اختصاص داده می شود. مقدار انتقال داده بین کارها قبلاً تعیین شده است. بنابراین، برای شروع محاسبات یک وظیفه، باید تمام داده های مورد نیاز برای آغازگر آن دریافت و آماده باشد.

۴.۳. روش پیشنهادی

روش پیشنهادی از خوشه بندی سطحی و افقی با درجه بندی هر وظیفه صورت می گیرد. در ادامه روش پیشنهادی را با بیان می کنیم.

در روش پیشنهادی ابتدا گراف جریان کاری بصورت افقی سطح بندی می شود. در هر سطح وظایف برحسب رتبه روبه بالایی که قبلا محاسبه شده است مرتب می شوند. این رتبه از وظیفه خروجی آغاز می شود و تا وظیفه ریشه ادامه می یابد. سپس وظایف انتخابی در هر سطح در یک گروه خوشه بندی می شوند. وظایف با محدودیت بیشینه درجه وابستگی گره، در گراف جریان کاری محاسبه می شوند. که بیانگر مجموع زمان ارسال داده (از والدین به وظیفه فعلی و از وظیفه فعلی به فرزندان) و زمان اجرای وظیفه می باشد. هر چه درجه وظیفه انتخاب شده کمتر باشد، تأثیر آن بر سایر وظایف کمتر است. در ادامه درجه همه وظایف در گراف جریان کاری محاسبه می شود و رتبه بندی روبه پایین سطحی آغاز می شود. ابتدا برای هر گراف از وظیفه ورودی تا وظیفه خروجی برای هر وظیفه در هر سطح یک رتبه ایجاد می شود. در گام بعدی وظایف بصورت نزولی رتبه در هر سطح مرتب می شوند. بعد از رتبه بندی همه وظایف گراف، وظایف در هر سطح از بالا به پایین براساس رتبه بصورت نزولی مرتب می شوند و در یک لیست قرار می گیرند. دلیل اینکار ایجاد اولویت بالا برای وظایفی است که بیشترین محدودیت را برای وظایف دیگر ایجاد می کنند. با توجه به اینکه هیچ وابستگی بین وظایفی که سطح یکسانی را اشغال می کنند وجود ندارد و بنابراین به صورت موازی قابل اجرا هستند و به طور همزمان در لیست خوانده شده قرار می گیرند.

سپس در ادامه ابتدا تعداد مجموعه وظایف متوالی که بصورت عمودی نشان داده شده اند، مطابق با رابطه والد و فرزند شمارش می شوند. از آنجا که مجموعه وظایف متوالی متعددی در جریان کاری وجود دارد، از روش رتبه صعودی برای سفارش هر یک از مجموعه وظایف متوالی برای اولویت پردازش استفاده می شود. در ادامه وظایف مستقلی که باقی مانده اند در هر مجموعه وظایف متوالی بصورت عمودی ادغام می شوند، به طوری که هزینه های ارتباطی کاهش می یابد. سپس خوشه ها برحسب بحرانی بودن مرتب می شوند. خوشه های بحرانی به ابرهایی تخصیص داده می شوند که مهلت زمانی را رعایت کنند. در ادامه زیرمهلت فرعی هر وظیفه براساس توزیع مهلت زمانی اصلی جریان کاری به دست می آوریم و با افزایش ترتیب مهلت های وظیفه، یک صف اولویت وظیفه برای وظایف آماده اجرای هر سطح ایجاد می شود. به این ترتیب، یک وظیفه با مهلت فرعی کوچک تر در پایین ترین اولویت، اولویت بندی می شود و برای یک وظیفه با مهلت فرعی بزرگ تر اولویت بالاتری تعیین می شود. بنابراین یک صف اولویت وظایف آماده با مهلت های فرعی وظیفه به دست می آید. در خوشه بندی عمودی هر وظیفه از صف اولویت آماده انتخاب و بهینه ترین منبع تخصیص می یابد. برای جستجوی منابع بهینه از روش جستجوی عمقی هوشمند گراف در هر شبکه مش از ابرها انجام می شود. عمل جستجوی منابع بهینه در هر ابر انجام می شود در صورتی که هیچ یک از منابع مهلت زمانی را رعایت نکند عمل جستجوی عمقی گراف، در گراف شبکه ابر دیگری آغاز می شود.

میانگین زمان اجرای هر وظیفه طبق رابطه ۱ بر روی ابرهای چندگانه محاسبه می شود:

$$\text{Execution}_{\text{time}}(t_i) = \frac{MI(t_i)}{\text{MIPS}(\text{AVG}(\text{all } vm \in VM_{\text{cloudj}}))} \quad (1)$$

زمان اجرای وظیفه i بر روی یک ماشین مجازی j در ابر شماره c طبق رابطه ۲ محاسبه می شود:

$$\text{Execution}_{\text{time}}(t_i, vm_j) = \frac{MI(t_i)}{\text{MIPS}(vm_{\text{cloudj}})} \quad (2)$$

زمان انتقال داده از ماشین مجازی i در ابر C به ماشین مجازی j در ابر k طبق رابطه ۳ محاسبه می شود:

$$\text{transfer_time}(vm_{\text{Cloudi}}, vm_{\text{kj}}) = \frac{\sum_{F \in \text{input files}} \text{size}(F)}{\text{bw}(vm_{\text{Cloudi}}, vm_{\text{cloudj}})} \quad (3)$$

زودترین زمان آغاز به اجرای یک وظیفه بر روی ماشین مجازی طبق رابطه ۴ محاسبه می‌شود:

$$EST(t_i) = \max_{t_p \in t_i \text{ parents}} \begin{cases} EST(t_p) + execution_{time}(t_p) + transfer_{time}(t_i) \\ EST(t_{in}) = 0 \end{cases} \quad (4)$$

زودترین زمانی که وظیفه کار خود را به اتمام می‌رساند طبق رابطه ۵ محاسبه می‌شود:

$$EFT(t_i) = EST(t_i) + execution_{time}(t_i) \quad (5)$$

دیرترین زمانی که وظیفه می‌توان کار خود را به اتمام برساند طبق رابطه ۶ محاسبه می‌شود:

$$Rank_u(t_i) = \begin{cases} grade_i + \max_{t_j \in succ_{ti}} (grade_j + Rank_u(t_j)) & \text{if } i \neq exit \\ grade_{t_{exit}} & \text{else} \end{cases} \quad (6)$$

رتبه‌بندی گره‌های بادرجه بالا از پایین ترین سطح به بالا ترین سطح از رابطه ۷ استفاده می‌شود. رتبه هر گره برابر است با حاصل جمع بزرگترین رتبه فرزندان و مقدار درجه گره فعلی است.

$$Rank_u(t_i) = \begin{cases} grade_i + \max_{t_j \in succ_{ti}} (grade_j + Rank_u(t_j)) & \text{if } i \neq exit \\ grade_{t_{exit}} & \text{else} \end{cases} \quad (7)$$

برای محاسبه والد بحرانی هر گره از رابطه ۸ استفاده می‌شود. والد بحرانی هر گره برابر انتخاب والدی با بیشترین زمان اجرا و زمان انتقال داده است.

$$CP_i = \max_{t_p \in parents_i} (Rank_u(t_i)) \quad (8)$$

مهلت سراسری کل گراف جریان کاری به زیر مهلت‌های متفاوت SD^۱ برای هر سطح تقسیم می‌شود و طبق رابطه ۹ محاسبه می‌شود.

$$SD(I_k) = \frac{\sum_{t_i \in I_k} ET_{t_i} + \sum_{e_{i,j} \in child_{t_i}} TT_{e_{i,j}}}{\sum_{t_i \in T} ET_{t_i} + \sum_{e_{ci,kj} \in E} TT_{e_{ci,kj}}} \times Deadline \quad (9)$$

برای محاسبه درجه گره از رابطه زیر استفاده می‌شود:

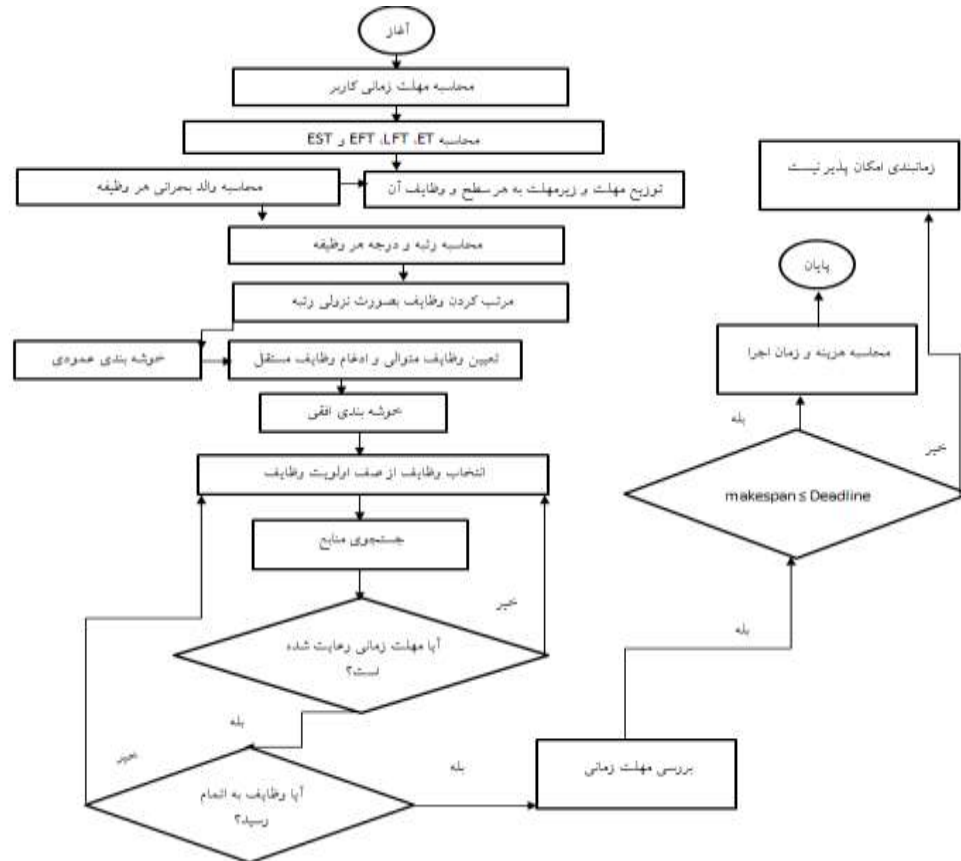
$$grade_i = (ET_i + \sum_{j=1, j \in parent_i}^m (ET_j + TT_{ji})) + \sum_{c=1, c \in child_i}^k (ET_c + TT_{ic}) \quad (10)$$

در رابطه ۱۰ متغیر m تعداد والد‌های گره i و k تعداد فرزندان گره i می‌باشد.

¹ Sub Deadline

۵.۳. فلوچارت روش پیشنهادی

شکل ۲ فلوچارت الگوریتم پیشنهادی را نشان می‌دهد. در گام اول مهلت زمانی و مقادیر اولیه محاسبه می‌شود. در گام بعدی والد



شکل ۲: فلوچارت پیشنهادی

بحرانی وظایف محاسبه و مهلت زمانی به هر سطح و وظایف آن توزیع می‌شود. در گام بعدی برای اولویت گذاری وظایف رتبه‌بندی و درجه هر وظیفه محاسبه می‌شود. وظایف براساس نزولی رتبه مرتب می‌شوند. در ادامه وظایف بصورت عمودی خوشه‌بندی می‌شوند. سپس وظایف متوالی مشخص و وظایف مستقل ادغام می‌شوند. در گام بعدی خوشه‌بندی افقی انجام می‌شود و بهترین منبع برای تخصیص انتخاب می‌شود به طوریکه مهلت زمانی را رعایت کند. در انتها متغیرها محاسبه می‌شوند.

۶.۳. شبه کد روش پیشنهادی

شکل ۳ خوشه‌بندی افقی را نشان می‌دهد در سطر ۱ تا ۵ برای هر جریان کاری تعداد وظایف متوالی در هر سطر شمارش می‌شود سپس طبق رابطه‌های ۷ و ۸ و ۱۰ همه وظایف اولویت‌بندی و رتبه‌بندی می‌شوند. سپس در سطر ۶ تا ۱۱ بای همه وظایف متوالی کمترین زمان اجرا تخمین زده می‌شود سپس رتباط بین والد و فرزند تمام خوشه‌های عمودی محاسبه می‌شود و در ادامه وظایف مستقل شناسایی می‌شوند و در یک خوشه قرار می‌گیرند. در ۱۲ زمان اجرای کلی محاسبه می‌شود و در سطر ۱۳ زمان‌بندی به روز رسانی می‌شود.

Function Vertical clustering with time consideration

1. For $k=1$ to N workflow do
2. Enumerate number of sequential task set
3. Compute and rank processing priority all sequential Task with EQ 7,8,10
4. Reserve the priority results of rank to the set Rank
5. End for
6. For $i=1$ to SeqT do
7. Total_ET=Estimation minimizes the total ET for $ET(t_j, VM_j)$ with in the maximum completion in sequential Task
8. Check the relationship of tasks from parent and child
9. Merge independ tasks
10. Continue check the next sequential taskset
11. End for
12. Compute the global average execution time with EQ 1
13. Schedule =Update Schedule [k]

end

شکل ۳: خوشه‌بندی عمودی

شکل ۴ خوشه‌بندی افقی را نشان می‌دهد در سطر ۱ مهلت فرعی وظایف هر سطح محاسبه می‌شود. در سطر ۲ تا ۶ برای هر جریان کاری تعداد وظایف موازی در هر سطر شمارش می‌شود سپس طبق رابطه‌های ۷ و ۸ و ۱۰ همه وظایف اولویت‌بندی و رتبه‌بندی می‌شوند. سپس در سطر ۷ تا ۸ منابع با کمترین هزینه برای وظایف هر سطح انتخاب می‌شوند. و در سطر ۱۲ زمانبندی جریان کاری k به روز رسانی می‌شود.

Function Horizontal clustering and cost minimization consideration

1. Calculate SD with EQ9
 2. For $k=1$ to N workflow do
 3. Enumerate number of parallel task set levels
 4. Compute and rank processing priority all Task leves with EQ 7,8,10
 5. Reserve the priority results of rank to the set Rank
 6. End for
 7. For $i=1$ to l do
 8. Total_cost=Estimation minimizes the total cost for $ET(t_j, VM_j)$ with in the maximum completion in level i
 14. End for
 15. Schedule =Update Schedule [k]
- end

شکل ۴: خوشه‌بندی افقی

در این بخش روش پیشنهادی ارائه شد. رویکرد زمان‌بندی پیشنهادی با آگاهی از عدم قطعیت جریان کاری‌های بلادرنگ مبتنی بر خوشه‌بندی سطحی افقی تحت محدودیت مهلت در سیستم‌های چند ابری، زمانبندی را انجام می‌دهد. بطوریکه هدف این الگوریتم

هزینه اجاره ماشین‌های مجازی جهت اجرای برنامه‌های جریان کاری واقعی، هزینه اجاره ماشین‌های مجازی در مهلت زمانی معین، بهبود نرخ نقض مهلت و کارایی استفاده از منابع ماشین مجازی است.

۴. یافته‌ها

در این بخش ابتدا تنظیمات شبیه‌سازی انجام شده سپس به تحلیل و ارزیابی نتایج پرداخته می‌شود.

۱.۴. تنظیمات شبیه‌سازی

برای ارزیابی عملکرد الگوریتم پیشنهادی، از روش پایه RWSMC (Xu et al, 2023) برای مقایسه استفاده می‌شود که هر دو الگوریتم زمانبندی جریان کاری هستند. ارزیابی عملکرد الگوریتم‌ها براساس هزینه اجاره منبع، زمان اجرای جریان کاری، درصد نقض مهلت و کارایی استفاده ماشین مجازی می‌باشد.

به عنوان پلتفرم آزمایشی، از یک لپ تاپ با پردازنده ۲.۸۰ گیگاهرتزی Core i۳ و حافظه ۳۲ گیگابایتی استفاده می‌شود و آزمایش‌های مقایسه‌ای توسط متلب ۲۰۲۰ اجرا می‌شوند. سیستم‌های چند ابری از سه ارائه‌دهنده ابر، یعنی مایکروسافت آژور، آمازون EC2 و موتور محاسباتی گوگل تشکیل شده‌اند. برای نشان دادن قابلیت‌های محاسباتی مختلف ماشین‌های مجازی، را به عنوان نسبت بالاترین قابلیت محاسباتی در سیستم به توانایی محاسباتی را داشته باشد.

در آزمایش‌های شبیه‌سازی، دو مجموعه داده می‌سازیم. کی از آنها مجموعه داده جریان کار واقعی است. این مجموعه داده شامل چهار نوع از جریان کاری‌های Montage، CyberShake، LIGO و SIPHT است. تعداد وظایف جریان کاری (۱۰۰۰، ۱۵۰۰، ۲۰۰۰، ۲۵۰۰ و ۳۰۰۰) است.

جدول ۱ مشخصات منابع مایکروسافت آژور را نشان می‌دهد، از جمله قیمت‌ها و قابلیت محاسباتی آن‌ها است. زمان بوت روی ۹۷ ثانیه، میانگین پهنای باند شبکه ۱۰۰ مگابیت بر ثانیه تنظیم شده است. برای همان ارائه دهنده ابر و ۲۰ مگابیت در ثانیه بین ارائه دهندگان خدمات ابری مختلف، یعنی ۱۰۰ مگابیت در ثانیه، ۲۰ مگابیت در ثانیه. قیمت منابع برحسب دقیقه و دلار در نظر گرفته می‌شود.

جدول ۱: مشخصات مایکروسافت آژور

نوع ماشین مجازی	قیمت منابع برحسب دقیقه (\$)	$R(vm_{i,j})$
B2MS	۰.۰۱۵	۸
B4MS	۰.۰۳	۴
B8MS	۰.۰۶	۲
B16MS	۰.۱۲	۱

جدول ۲ مشخصات منابع آمازون EC2 را نشان می‌دهد. قیمت منابع برحسب ساعت و دلار در نظر گرفته می‌شود.

جدول ۲: آمازون EC2

نوع ماشین مجازی	قیمت منابع برحسب ساعت (\$)	$R(vm_{i,j})$
m1.small	۰.۰۶	۸
m1.medium	۰.۱۲	۴
m1.large	۰.۲۴	۲
m1.xlarge	۰.۴۵	۱

جدول ۳ مشخصات منابع موتور محاسباتی گوگل را نشان می‌دهد. قیمت منابع برحسب یک دقیقه و ده دقیقه و دلار در نظر گرفته می‌شود.

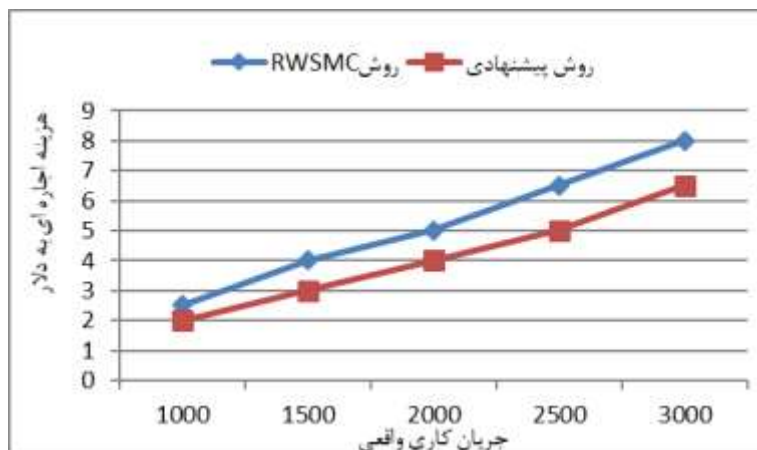
جدول ۳: موتور محاسباتی گوگل

نوع ماشین مجازی	قیمت منابع برحسب ده دقیقه (\$)	قیمت منابع برحسب دقیقه (\$)	$R(vm_{i,j})$
n1-highcpu-2	۰.۱۴	۰.۰۰۱۲	۸
n1-highcpu-4	۰.۲۵	۰.۰۰۲۳	۴
n1-highcpu-8	۰.۵	۰.۰۰۴۷	۲
n1-highcpu-16	۱	۰.۰۰۹۳	۱

۲.۴. نتایج حاصل از انجام شبیه‌سازی

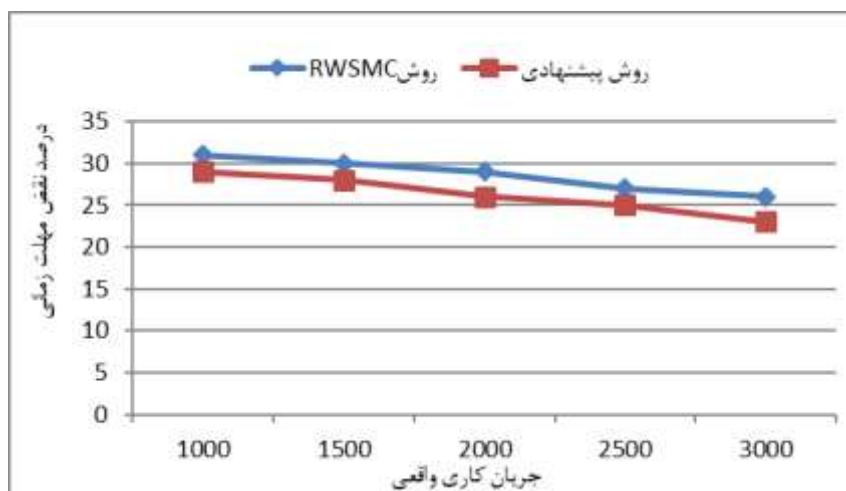
آزمایش را برای هر الگوریتم به طور مستقل ۱۰ بار تکرار می‌کنیم و میانگین نتایج را برای مقایسه عملکرد آن‌ها به دست می‌آوریم. ابتدا آزمایش‌ها عملکرد الگوریتم‌ها را در سه معیار نشان می‌دهند که اندازه مجموعه داده‌های جریان کار واقعی در محدوده ۱۰۰۰ تا ۳۰۰۰ تغییر می‌کند و سایر پارامترها را ثابت می‌کند.

شکل ۱ هزینه اجرای جریان‌کاری‌های واقعی در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، هزینه الگوریتم‌ها افزایش می‌یابد زیرا با افزایش تعداد جریان‌کاری، الگوریتم‌ها نیاز به اجاره ماشین‌های مجازی بیشتری برای اجرای وظایف دارند. وقتی تعداد جریان‌کاری‌های واقعی افزایش می‌یابد، میانگین هزینه اجاره به دست آمده در روش پایه برابر ۵.۲ دلار و در روش پیشنهادی ۴.۱ دلار است. نتایج نشان می‌دهد که الگوریتم پیشنهادی هزینه کمتری دارد.



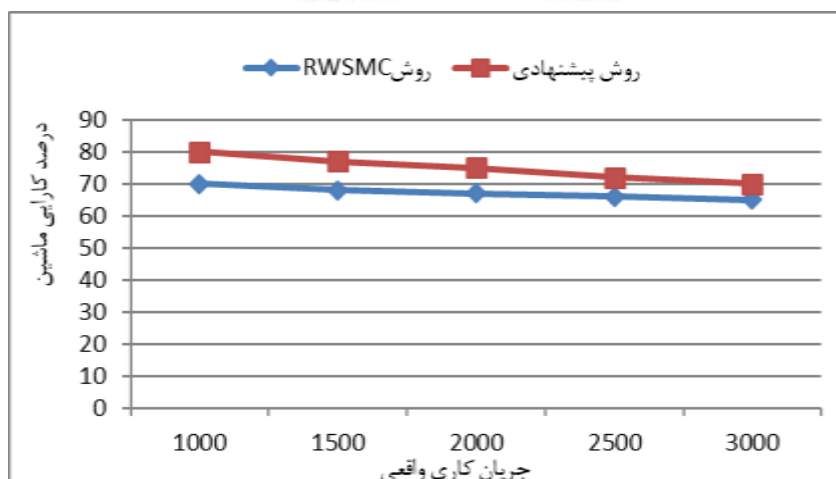
شکل ۱: هزینه اجرای جریان کاری‌های واقعی در روش پیشنهادی و RWSMC

شکل ۲ درصد نقض مهلت زمانی اجرای جریان کاری‌های واقعی در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، نقض مهلت زمانی الگوریتم‌ها کاهش می‌یابد زیرا با افزایش تعداد جریان کاری، الگوریتم‌ها نیاز به اجرای ماشین‌های مجازی بیشتری برای اجرای وظایف دارند. وقتی تعداد جریان کاری‌های واقعی افزایش می‌یابد، میانگین نقض مهلت به دست آمده در روش پایه برابر ۲۸.۶ و در روش پیشنهادی ۲۶.۲ است. نتایج نشان می‌دهد که الگوریتم پیشنهادی نقض مهلت زمانی کمتری دارد.



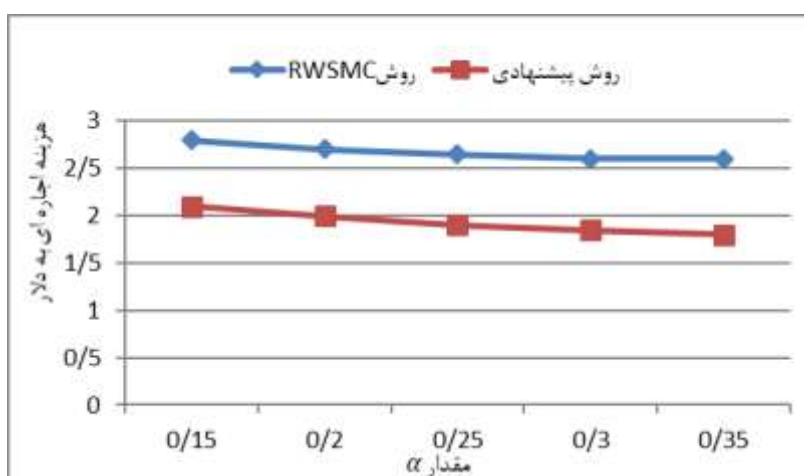
شکل ۲: درصد نقض مهلت زمانی اجرای جریان کاری‌های واقعی در روش پیشنهادی و RWSMC

شکل ۳ درصد کارایی ماشین مجازی اجرای جریان کاری‌های واقعی در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، کارایی ماشین مجازی الگوریتم‌ها کاهش می‌یابد زیرا با افزایش تعداد جریان کاری، الگوریتم‌ها نیاز به اجرای حجم بیشتری از کارها بر روی منابع هستند و کارایی ماشین مجازی کاهش می‌یابد. وقتی تعداد جریان کاری‌های واقعی افزایش می‌یابد، میانگین کارایی ماشین مجازی به دست آمده در روش پایه برابر ۶۷.۲ و در روش پیشنهادی ۷۴.۸ است. نتایج نشان می‌دهد که الگوریتم پیشنهادی درصد کارایی منابع را افزایش می‌دهد.



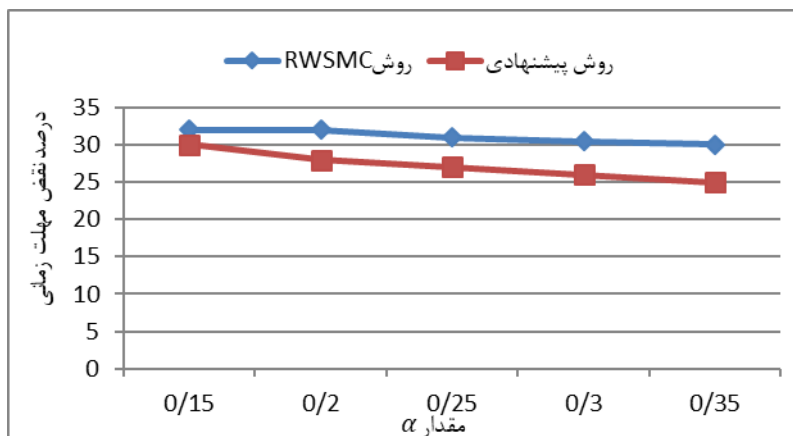
شکل ۳: درصد کارایی ماشین مجازی اجرای جریان کاری های واقعی در روش پیشنهادی و RWSMC

شکل ۴ هزینه اجاره اجرای جریان کاری های واقعی با مقدار α در بازه ۰.۱۵ تا ۰.۳۵. در روش پیشنهادی و RWSMC را نشان می دهد. با افزایش اندازه مجموعه داده، هزینه الگوریتم ها افزایش می یابد زیرا با افزایش تعداد جریان کاری، الگوریتم ها نیاز به اجاره ماشین های مجازی بیشتری برای اجرای وظایف دارند. وقتی تعداد جریان کاری های واقعی افزایش می یابد، میانگین هزینه اجاره به دست آمده در روش پایه برابر ۲.۶۷ دلار و در روش پیشنهادی ۱.۹۳ دلار است. نتایج نشان می دهد که الگوریتم پیشنهادی هزینه کمتری دارد.



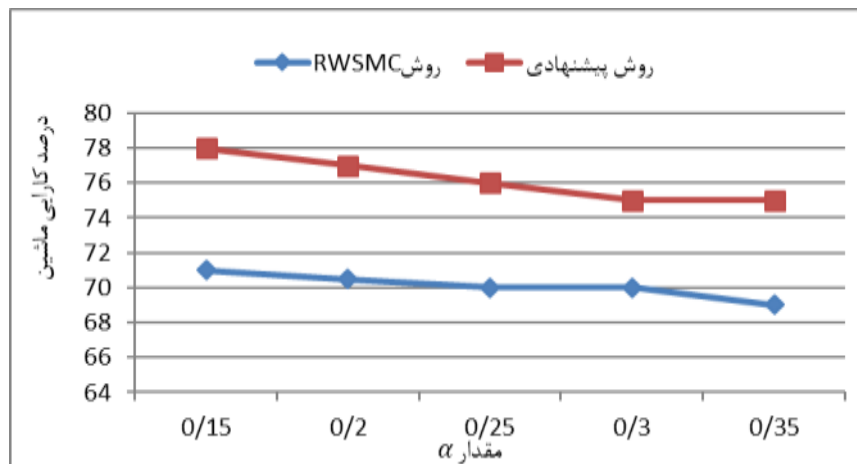
شکل ۴: هزینه اجاره در زمان اجرای کار با مقدار α

شکل ۵ درصد نقض مهلت زمانی اجرای جریان کاری های واقعی با مقدار α در بازه ۰.۱۵ تا ۰.۳۵. در روش پیشنهادی و RWSMC را نشان می دهد. با افزایش اندازه مجموعه داده، نقض مهلت زمانی الگوریتم ها کاهش می یابد زیرا با افزایش تعداد جریان کاری، الگوریتم ها نیاز به اجاره ماشین های مجازی بیشتری برای اجرای وظایف دارند. وقتی تعداد جریان کاری های واقعی افزایش می یابد، میانگین نقض مهلت به دست آمده در روش پایه برابر ۳۱.۱ و در روش پیشنهادی ۲۷.۲ است. نتایج نشان می دهد که الگوریتم پیشنهادی نقض مهلت زمانی کمتری دارد.



شکل ۵: درصد نقض مهلت زمانی در زمان اجرای کار با مقدار α

شکل ۶ درصد کارایی ماشین مجازی اجرای جریان کاری‌های واقعی با مقدار α در بازه ۰.۱۵ تا ۰.۳۵. در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، کارایی ماشین مجازی الگوریتم‌ها کاهش می‌یابد زیرا با افزایش تعداد جریان کاری، الگوریتم‌ها نیاز به اجرای حجم بیشتری از کارها بر روی منابع هستند و کارایی ماشین مجازی کاهش می‌یابد. وقتی تعداد جریان کاری‌های واقعی افزایش می‌یابد، میانگین کارایی ماشین مجازی به دست آمده در روش پایه برابر ۷۰.۱ و در روش پیشنهادی ۷۶.۲ است. نتایج نشان می‌دهد که الگوریتم پیشنهادی درصد کارایی منابع را افزایش می‌دهد.

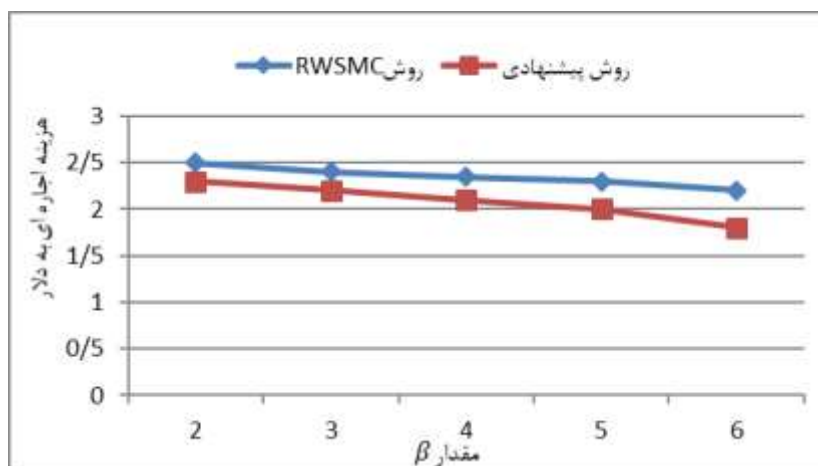


شکل ۶: درصد کارایی ماشین مجازی در زمان اجرای کار با مقدار α

در ادامه عملکرد هر الگوریتم را بررسی می‌کنیم که تحت مقادیر مختلف β تغییر کرده و سایر پارامترها ثابت شده‌اند. از آنجایی که هر الگوریتم وظایف زمان‌بندی را برای ماشین‌های مجازی با قابلیت محاسباتی ارزان‌تر و پایین‌تر برای به دست آوردن کمترین هزینه اجاره، اولویت‌بندی می‌کند. نتایج نشان می‌دهد که هزینه کل اجاره هر چهار الگوریتم با افزایش ضریب مهلت β از ۲ به ۶ کاهش می‌یابد.

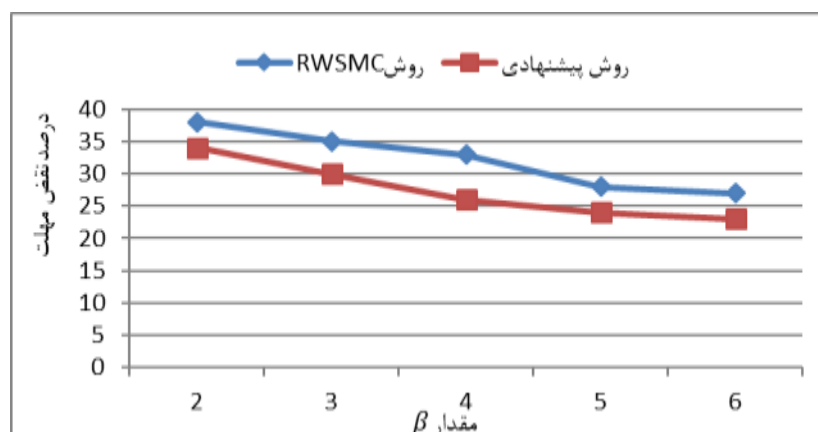
شکل ۷ هزینه اجاره اجرای جریان کاری‌های واقعی با مقدار β در بازه ۲ تا ۶ در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، هزینه الگوریتم‌ها افزایش می‌یابد زیرا با افزایش تعداد جریان کاری، الگوریتم‌ها نیاز به اجاره ماشین‌های مجازی

بیشتری برای اجرای وظایف دارند. وقتی تعداد جریان کاری‌های واقعی افزایش می‌یابد، میانگین هزینه اجاره به دست آمده در روش پایه برابر ۲.۳۵ دلار و در روش پیشنهادی ۲.۸ دلار است. نتایج نشان می‌دهد که الگوریتم پیشنهادی هزینه کمتری دارد.



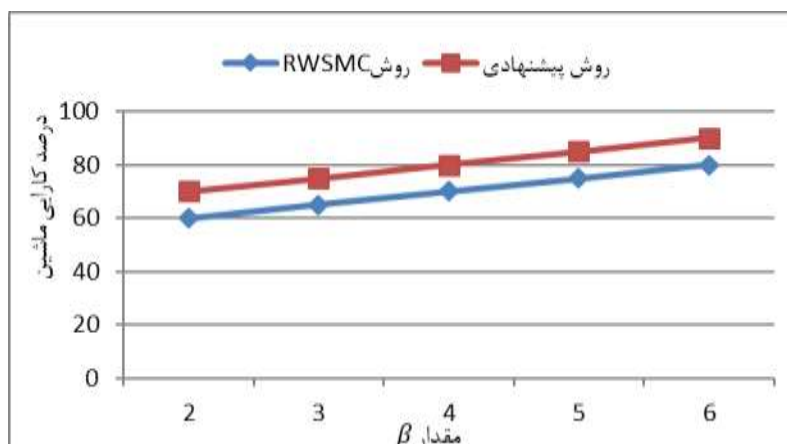
شکل ۷: هزینه اجاره در زمان اجرای کار با مقدار β

شکل ۸ درصد نقض مهلت زمانی اجرای جریان کاری‌های واقعی با مقدار β در بازه ۲ تا ۶ در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، نقض مهلت زمانی الگوریتم‌ها کاهش می‌یابد زیرا با افزایش تعداد جریان کاری، الگوریتم‌ها نیاز به اجاره ماشین‌های مجازی بیشتری برای اجرای وظایف دارند. وقتی تعداد جریان کاری‌های واقعی افزایش می‌یابد، میانگین نقض مهلت به دست آمده در روش پایه برابر ۳۲.۲ و در روش پیشنهادی ۲۷.۴ است. نتایج نشان می‌دهد که الگوریتم پیشنهادی نقض مهلت زمانی کمتری دارد.



شکل ۸: درصد نقض مهلت زمانی در زمان اجرای کار با مقدار β

شکل ۹ درصد کارایی ماشین مجازی اجرای جریان کاری‌های واقعی با مقدار β در بازه ۲ تا ۶ در روش پیشنهادی و RWSMC را نشان می‌دهد. با افزایش اندازه مجموعه داده، کارایی ماشین مجازی الگوریتم‌ها کاهش می‌یابد زیرا با افزایش تعداد جریان کاری، الگوریتم‌ها نیاز به اجرای حجم بیشتری از کارها بر روی منابع هستند و کارایی ماشین مجازی کاهش می‌یابد. وقتی تعداد جریان کاری‌های واقعی افزایش می‌یابد، میانگین کارایی ماشین مجازی به دست آمده در روش پایه برابر ۷۰ و در روش پیشنهادی ۸۰ است. نتایج نشان می‌دهد که الگوریتم پیشنهادی درصد کارایی منابع را افزایش می‌دهد.



شکل ۹: درصد کارایی ماشین مجازی در زمان اجرای کار با مقدار β

۵. بحث و نتیجه گیری

در این مقاله زمان بندی با آگاهی از عدم قطعیت جریان کاری های بلادرنگ مبتنی بر خوشه بندی سطحی افقی تحت محدودیت مهلت در سیستم های چند ابری ارائه شده است. روش پیشنهادی از یک روش خوشه بندی افقی و عمودی استفاده می کند. خوشه بندی افقی هزینه را کاهش می دهد و خوشه بندی عمودی زمان اجرا را کاهش می دهد. در روش پیشنهادی ابتدا گراف جریان کاری بصورت افقی سطح بندی می شود. در هر سطح وظایف برحسب رتبه روبه بالایی که قبلا محاسبه شده است مرتب می شوند. سپس وظایف انتخابی در هر سطح در یک گروه خوشه بندی می شوند. گره های با محدودیت بیشینه درجه وابستگی گره، در گراف جریان کاری محاسبه می شوند. سپس در ادامه ابتدا تعداد مجموعه وظایف متوالی که بصورت عمودی برای کاهش زمان اجرا مشخص می شود سپس از روش رتبه صعودی برای سفارش هر یک از مجموعه وظایف متوالی برای اولویت پردازش استفاده می شود. در ادامه وظایف مستقلی که باقی مانده اند در هر مجموعه وظایف متوالی بصورت عمودی ادغام می شوند، به طوری که هزینه های ارتباطی کاهش می یابد. هدف الگوریتم پیشنهادی به حداقل رساندن هزینه اجرای وظایف با رعایت محدودیت مهلت زمانی جریان کاری است. متغیرهای هزینه، زمان اجرا درصد نقض مهلت و کارایی استفاده ماشین مجازی برای هر دو الگوریتم ارزیابی شدند. نتایج نشان داد که روش پیشنهادی عملکرد بهتری نسبت به روش RWSMC دارد.

برای کارهای آینده، مکانیسم های صورت حساب پیچیده تری مانند نمونه های درخواستی، نمونه های نقطه ای، و نمونه های چانه زنی را معرفی خواهیم کرد که در نتیجه هزینه های اجرای کمتری به همراه خواهد داشت. بررسی قابلیت اطمینان و تحمل خطا در انتقال داده در محیط شبکه چند ابری از دیگر مواردی است که انجام خواهد شد. دورنمای زمان بندی جریان کاری برای کاهش مصرف انرژی در مراکز داده ابری متعدد را با روش فرا ابتکاری مطالعه خواهیم کرد.

منابع

- Jin Xu, Huiqun Yu, Guisheng Fan, Jiayin Zhang(2023), "Uncertainty-aware scheduling of real-time workflows under deadline constraints on multi-cloud systems Concurrency Computa" Pract Exper.35:e7562
- Armbrust M, Fox A, Griffith R, et al.(2010),"A view of cloud computing". Commun ACM;53(4):50-58.
- Masdari M, ValiKardan S, Shahi Z, Azar SI.(2016),"Towards workflow scheduling in cloud computing: a comprehensive analysis". J Netw Comput Appl;66:64-82.

- Dogan A, Ozguner F.(2002)," Matching and scheduling algorithms for minimizing execution time and failure probability of applications in heterogeneous computing". IEEE Trans Parallel Distribut Syst;13(3):308-323.
- HassanHA, Salem SA, Saad EM.(2020),"A smart energy and reliability aware scheduling algorithm for workflow execution inDVFS-enabled cloud environment". Futur Gener Comput Syst;112:431-448.
- Xie G, Zeng G, Li R, Li K.(2017),"Quantitative fault-tolerance for reliable workflows on heterogeneous IaaS clouds. IEEE Trans Cloud Comput";8(4):1223-1236.
- Xueyang Chen,(2023),"Multi-objective optimization task scheduling method based on dynamic programming for multi-cloudenvironment International Conference on Information Science", Parallel and Distributed Systems(ISPDS)|979-8-3503-3718-1/23©IEEE|DOI:10.1109/ISPDS58840.10235565z
- Abrishami S, Naghibzadeh M, Epema DH.(2013)," Deadline-constrained workflow scheduling algorithms for infrastructure as a service clouds". Futur Gener Comput Syst;29(1):158-169
- Juve G, Chervenak A, Deelman E, Bharathi S, Mehta G, Vahi K.(2013),"Characterizing and profiling scientific workflows". Futur Gener Comput Syst;29(3):682-692.
- WuQ, Ishikawa F, Zhu Q, Xia Y,Wen J.(2017),"Deadline-constrained cost optimization approaches for workflow scheduling in clouds". IEEE Trans ParallelDistribut Syst;28(12):3401-3412.
- Alkhanak EN, Lee SP, Rezaei R, Parizi RM.(2016),"Cost optimization approaches for scientific workflow scheduling in cloud and grid computing: A review, classifications, and open issues". J Syst Softw;113:1-26.
- ZhangR, ShiW.(2021),"Amakespan-optimized task-Level scheduling strategy for cloud workflow systems". International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT). IEEE; 712-720.
- Cai Z, Li X, Ruiz R.(2017),"Resource provisioning for task-batch based workflows with deadlines in public clouds". IEEE Trans Cloud Comput. 2017;7(3):814-826.
- Ding R, Li X, Liu X, Xu J.(2018),"A cost-effective time-constrained multi-workflow scheduling strategy in fog computing". International Conference on Service-Oriented Computing. Springer;194-207.
- ArabnejadV,BubendorferK,NgB.(2019),"Dynamic multi-workflow scheduling:Adeadline and cost-aware approach for commercial clouds". FuturGener Comput Syst;100:98-108.
- Liu J, Ren J, Dai W, et al.(2021),"Online multi-workflow scheduling under uncertain task execution time in IaaS clouds". IEEE Trans Cloud Comput;9(3):1180-1194. doi:10.1109/TCC.2906300
- Ma X, Xu H, Gao H, BianM.(2021),"Real-time multiple-workflow scheduling in cloud environments". IEEE Trans Netw ServManag;18(4):4002-4018. doi:10. 1109/TNSM.3125395
- Ahmad Taghinezhad-Niar, Saeid Pashazadeh, Javid Taheri,(2022),"Energy-efficient workflow scheduling with budget-deadline constraints for cloud" <https://doi.org/10.1007/s00607-021-01030-9> Computing,104:601–625
- K. Kalyana Chakravarthi, P. Neelakantan, L. Shyamala, V. Vaidehi, (2022),"Reliable budget aware workflow scheduling strategy on multi-cloud environment",<https://doi.org/10.1007/s10586-021-03464-4>(0123456789,volV)(0123456789,volV) Cluster Computing 25:1189–1205
- Arash Deldari, Abolghasem Yousofi, Mahmoud Naghibzadeh, Alireza Salehan(2022)," CDA: a novel multicore scheduling for cost-aware deadline-constrained scientific workflows on the IaaS cloud", <https://doi.org/10.1007/s11227-022-04551-y> The Journal of Supercomputing 78:17027–17054
- Xiao Ge, Qiang Zhang,(2022),"Efficient Computation Offloading with Energy Consumption Constraint for Multi-Cloud System" 15th International Conference on Advanced Computer Theory and Engineering (ICACTE) | 978-1-6654-8180-9/22 IEEE | DOI: 10.1109/ICACTE55855.9943608
- Xiaoyong Tang,(2022),"Reliability-Aware Cost-Efficient Scientific Workflows Scheduling Strategy on Multi-Cloud Systems"IEEE TRANSACTIONS ON CLOUD COMPUTING, VOL. 10, NO. 4, OCTOBER-DECEMBER
- Weifan Zhang, Sokol Kosta, Preben Mogensen,(2023),"A Scheduling Method for Tasks and Services in IIoT Multi-Cloud Environments", International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT) | 979-8-3503-4649-7/23 ©IEEE
- SAMUEL MANAM, KLAUS MOESSNER,SERDAR VURAL,(2023),"Deadline-Constrained Cost Minimisation for Cloud Computing Environments", Digital Object Identifier 10.1109/ACCESS.3258682 IEEE
- Pradhan, Roshni, Satapathy, Suresh Chandra,(2022)," Energy-Aware Cloud Task Scheduling algorithm in heterogeneous multi-cloud environment", Journal: Intelligent Decision Technologies, vol. 16, no. 2, pp. 279-284.
- Mazen Farid, Heng Siong Lim, Chin Poo Lee and Rohaya Latip,(2023),"Scheduling ScientificWorkflow in Multi-Cloud:A Multi-Objective Minimum Weight Optimization Decision-Making Approach" Symmetry, 15, 2047. <https://doi.org/10.3390/sym15112047>
- Amazon EC2. Accessed 01 March,(2022). <https://aws.amazon.com/ec2/>.
- Google Compute Engine. Accessed 01 March,(2022). <https://cloud.google.com/compute>.
- Microsoft Azure. Accessed 01 March, (2022). <https://azure.microsoft.com>.

Uncertainty-Aware Scheduling of Real-Time Workflows Based on Horizontal Surface Clustering Under Deadline Constraints in Multi-Cloud Systems

Marzieh Sadat Afzali

Master's degree student in Information Technology Engineering, E-Commerce major, Atrak Quchan Institute of Higher Education

Dr. Masoud Khosravi Pharmad

Lecturer in the Department of Information Technology Engineering, Atrak Quchan Institute of Higher Education

Abstract

Workflow scheduling in multi-cloud systems has also attracted the attention of many researchers. However, it is challenging due to the heterogeneous resource types and multiple billing mechanisms. In recent years, cloud computing technology has been widely used and widely deployed. Therefore, one of the issues to be solved in the development of cloud computing is to assign tasks to appropriate virtual machines and reduce the idle waiting time of rented virtual machines to obtain the optimal rental cost. In this paper, an uncertainty-aware scheduling algorithm for real-time workflows based on horizontal surface clustering under deadline constraints is presented. The proposed method uses a horizontal and vertical clustering method. Horizontal clustering reduces the cost and vertical clustering reduces the execution time. The proposed algorithm aims to minimize the rental cost of executing tasks while observing the deadline constraint of the workflow. So that the percentage of deadline violations decreases and the efficiency of virtual machine usage increases. Simulations were performed in the MATLAB environment for the proposed method and the baseline method for the variables of cost, execution time, percentage of deadline violations, and efficiency of virtual machine usage. Both algorithms were evaluated and the results showed that the proposed method has better performance than the baseline method.

Keywords: Multi cloud scheduling, uncertainty, horizontal surface clustering, real-time.