

ارائه یک الگوریتم بهینه برای زمان بندی منابع و پارتیشن بندی کد در محاسبات ابری تلفن همراه

چکیده

با تکنولوژی مجازی سازی، مراکز داده ابری امروزی انعطاف پذیرتر، امن تر می شود و بر اساس تقاضا تخصیص می یابد. یک فن آوری کلیدی که نقش مهمی در مراکز داده ابری بازی می کند، برنامه زمان بندی منابع است. در این مقاله برای حل مشکلات موجود در این حوزه با استفاده از الگوریتم تکاملی ازدحام ذرات برای کوچک کردن حیطه اهداف چندگانه به اندازه ای مناسب، راهبرد جایابی نزدیک به بهینه ارائه شده است. روش جایابی مبتنی بر الگوریتم بهینه سازی ازدحام ذرات می تواند به عنوان یک روش جایابی زمان واقعی به دلیل افزایش در قابلیت محاسباتی پردازشگرها در پنج سال گذشته، عمل کند. این جایابی یک روش جستجو بوده که در آن شایستگی به صورتی پویا بر مبنای واریانس های مقادیر تناسب در هر نسل تغییر می کنند. این روش مهاجرت و جایابی، به حداقل رسانی زمان تکمیل ماشین های مجازی نیز توجه می کند. برای ارزیابی روش پیشنهادی، نتایج حاصله طی معیارهای متفاوت کیفیتی و از جنبه های متفاوت و بر اساس تغییر در پارامترهای مختلف اجرایی مورد مقایسه و بررسی قرار گرفت. کارایی روش معرفی شده در مقایسه با سایر روش های موجود در این زمینه مقایسه شد که نتایج بیانگر کیفیت بالای روش پیشنهادی می باشد.

کلمات کلیدی: محاسبات ابری تلفن همراه، زمان بندی منابع، پارتیشن بندی کد، بهینه سازی ازدحام ذرات، ماشین مجازی.

مقدمه

رایانش ابری امکان محاسبه، ارتباطات و منابع ذخیره سازی نامحدود را به صورت مجازی فراهم می کند؛ به طوری که این منابع بر اساس تقاضای مشتریان با پیروی از مدل تجاری "پرداخت در ازای استفاده" تأمین می شوند. به منظور رویارویی با افزایش تقاضاهای مصرف کننده، ارائه دهندگان ابر در حال استقرار مراکز داده ی بزرگ مقیاس شامل صدها هزار گره در سراسر دنیا می باشند مراکز داده ی بزرگ امروزی بستری برای میزبانی طیف وسیعی از برنامه های کاربردی ترکیبی هستند. در مدل محاسبات ابری استاندارد کاربر از منابع و سرویس های یک مرکز داده استفاده می کند. به دلیل تنوع ابرها، نمونه های ماشین مجازی و همچنین متفاوت بودن کارایی و قیمت آنها پیدا کردن ماشین مجازی مناسب برای کاربران یک مسئله پیچیده است. بنابراین به کارگیری مکانیزم کارگزاری منابع در رایانش ابری جهت تطبیق نیازهای کاربرد با منابع ابرها (نمونه های ماشین مجازی) به صورت شفاف از دیدگاه کاربر ضروری به نظر می رسد [۱].

تلاش برای دستیابی به کیفیت سرویس و صرفه جویی در مصرف انرژی یکی از مهم ترین موضوعاتی است که در مسئله زمان بندی منابع مورد توجه قرار گرفته است. برآورده کردن کیفیت سرویس در سطوح مختلف از سطح کسب و کار، طبقه بندی و راضی نگهداشتن مشتریان همچنین برآورده ساختن کیفیت سرویس در سطح جایگذاری ماشین های مجازی و پرداخت جریمه در صورت برآورده نشدن مهلت زمانی در تخصیص منابع مورد توجه می باشد و اهمیت این مسئله به خصوص با بالا رفتن استفاده از اینترنت موبایلی و افزایش یافتن کاربران موبایل افزایش یافته است. رایانش ابری موبایل به سه لایه تجزیه می شود: ابرهای عمومی، ابرچه ها و دستگاه های

تک کاره مانند دستگاه‌های بلوتوث. فاصله بیشتر بین کاربر و ابرچه انرژی و زمان بیشتری برای تخلیه بار مصرف می‌کند. سیگنال‌های رادیویی با پخش شدن در محیط به طور نمایی تضعیف می‌شوند، فاصله بیشتر بین دو جفت به معنای نرخ بیت مؤثر کمتر و کاهش نرخ انتقال است، در نتیجه زمان ارتباط افزایش می‌یابد. افزایش زمان ارتباط منجر به مصرف انرژی بیشتر می‌شود، بنابراین بهره‌وری ابرچه و طول صف انتظار باید در فرایند تصمیم‌گیری برای انتخاب ابرچه در نظر گرفته شود. این تصمیم‌گیری بدین منظور انجام می‌شود که میانگین زمان خاتمه تمام وظایف و مصرف برق باتری را کم کند [۲].

سه سناریو ممکن در شبکه موبایلی عبارتند از: ۱- تخلیه بار به یک ابرچه: که نزدیک‌ترین ابرچه با میزان بار کمتر یک انتخاب ایده آل برای کاربر است هنگامی که تعداد وظایف در انتظار برای اجرا در نزدیکی ابرچه افزایش یابد، به دلیل محدودیت منابع، کاربران تأخیر بیشتری را تجربه می‌کنند به هر حال این سناریو همیشه بهترین انتخاب نیست. ۲- تخلیه بار به یک ابر عمومی از طریق ابرچه‌ها: وقتی که تعداد وظایف بالا باشد، همه سرورها در ابرچه‌ها مشغول می‌شوند و صفوف طولانی از وظایف منتظر اجرا در ابرچه‌ها ایجاد می‌شود. ۳- تخلیه بار روی یک ابر عمومی از طریق اتصال 3G: که اگر تعداد درخواست‌های اضافه شده زیاد باشد، پهنای باند شلوغ می‌شود و از دست رفتن بسته‌ها و دوره عکس‌العمل زیاد زیاد می‌شود. با استفاده از الگوریتم ترکیبی ابتکاری مبتنی بر صف می‌توان بهترین مکان برای تخلیه بار وظایف موبایل را با در نظر گرفتن منابع در ابرچه‌ها و مدت ارتباط بین کاربران و ابرچه‌ها، پیدا کرد. مدت ارتباط به فاصله کاربر از ابرچه وابسته است. کاربر باید حداکثر زمان انتظار قابل تحمل را به عنوان یک پارامتر کیفیت مشخص کند [۳]. هدف اصلی الگوریتم پیشنهادی بهبود کارایی کلی سیستم با استفاده از کم کردن زمان خاتمه و مصرف برق باتری موبایل است. همچنین تلاش می‌کند تا تعداد درخواست‌های رد شده، بار ابرچه‌ها را موازنه کند. محاسبات ابری در دنیای امروز به خصوص در عرصه شبکه‌های موبایل نقش مهمی را بازی می‌کند و سرویس‌های متعددی را از طریق شبکه فراهم می‌نماید. ارائه‌دهندگان سرویس‌های ابری اغلب مراکز داده‌ی بزرگی را اداره می‌کنند. مراکز داده‌ی امروزی روز به روز در حال بزرگ‌تر و پیچیده‌تر شدن هستند. برنامه‌های کاربردی که توسط این مراکز میزبانی می‌گردند نیز در حال پیچیده‌تر شدن می‌باشند و هر یک طیف وسیع و متنوعی از نیازهای ارتباطی را دارا می‌باشند [۴]. ادامه این مقاله به این صورت سازماندهی خواهد شد که در قسمت اول پژوهش‌هایی که در گذشته در این حوزه پرداخته شده‌اند، بررسی خواهند شد و در ادامه روش پیشنهادی بیان خواهد شد، روش پیشنهادی پیاده‌سازی شده است و نتایج آن بیان می‌شود و در نهایت به بحث و نتیجه‌گیری پرداخته خواهد شد.

پیشینه پژوهش

با توجه به رشد روزافزون و سریع سیستم‌های محاسبات ابری موبایل، ارائه الگوریتم‌های جدید و کارآمد برای حل مسئله زمان‌بندی^۱ فرآیندها در این سیستم‌ها ضروری به نظر می‌رسد. گرید یک محیط عملیاتی شامل

¹ Schedule

انواع پردازنده‌ها با قابلیت‌های پردازشی گوناگون است که عملیات زمان‌بندی کارها و تخصیص منابع در آن را با دشواری‌هایی روبرو می‌کند. مدیریت منابع در محیط محاسبات ابری موبایل به علت توزیع جغرافیایی، ناهمگونی و پویا بودن منابع مسئله پیچیده‌ای است. یافتن بهترین پردازنده برای انجام هر فرآیند از میان تمام پردازنده‌های موجود با قابلیت‌های گوناگون کار پیچیده‌ای است، به طوری که بتوان طبق درخواست کاربر منبعی را پیدا کرد که بیشترین تطابق را با نیاز مطرح‌شده دارا باشد. در مسئله زمان‌بندی در محیط محاسبات ابری موبایل هدف یافتن یک روش مناسب برای انتساب فرآیندها به پردازنده‌های ناهمگون است [۵].

الگوریتم‌های زمان‌بندی در سیستم‌های کامپیوتری توزیع‌شده و موازی سنتی که معمولاً روی منابع اختصاصی و ناهمگن همچون محاسبات ابری موبایل اجرا می‌شوند، نمی‌توانند در شرایط و موقعیت‌های جدید به خوبی کار کنند. زمان‌بندی وظایف در محاسبات ابری موبایل بسیار پیچیده‌تر از زمان‌بندی منبع محلی است. چون باید منابع در مقیاس بزرگ‌تر را مدیریت و کنترل کند. روش‌های قدیمی زمان‌بندی برای مدیریت منابع در محاسبات ابری موبایل مناسب نیستند. هدف روش‌های قدیمی بالا بردن کارایی سیستم، افزایش بهره‌وری و برآورده کردن نیازهای منابع بود اما در روش‌های جدید کیفیت سرویس برای کاربران مهم است. هدف زمان‌بند تعادل بین هزینه و زمان اجرای کارها روی منابع مختلف می‌باشد [۶].

زمان‌بندی وظایف و تخصیص منابع در مراکز داده‌ای محاسبات ابری موبایل شامل سه مرحله اصلی می‌باشد: کشف منبع^۱، که لیستی از منابع ایجاد می‌کند، اطلاعات درباره آن منابع جمع‌آوری می‌شود. انتخاب یک مجموعه خوب از آن‌ها^۲. اجرای کار^۳ صورت می‌گیرد که شامل مرحله‌بندی و پاک‌سازی فایل‌های اضافی می‌باشد. یکی از مهم‌ترین چالش‌ها در زمان‌بندی وظایف ابری موبایل، توازن بار است. هدف از توازن بار، یافتن نگاشتی مناسب از کارها بر روی پردازنده‌های موجود در سیستم است به طوری که در هر پردازنده مقدار تقریباً مساوی از کارها اجرا گردد تا زمان اجرای کلی به کم‌ترین مقدار خود برسد. از آنجاکه مسئله توازن بار در کارایی سیستم‌ها بسیار تأثیرگذار است، توجه بسیاری از محققین را در سال‌های اخیر به خود معطوف کرده و کارهای زیادی در این زمینه انجام گرفته است و تلاش‌ها همچنان برای افزایش سرعت سیستم‌ها در انجام محاسبات و بهره‌گیری هر چه بیشتر از منابع در دسترس ادامه دارد. با داشتن شبکه‌هایی با سرعت بالا، تقاضاها برای به دست آوردن کارایی بالای محاسباتی در حال افزایش است [۷].

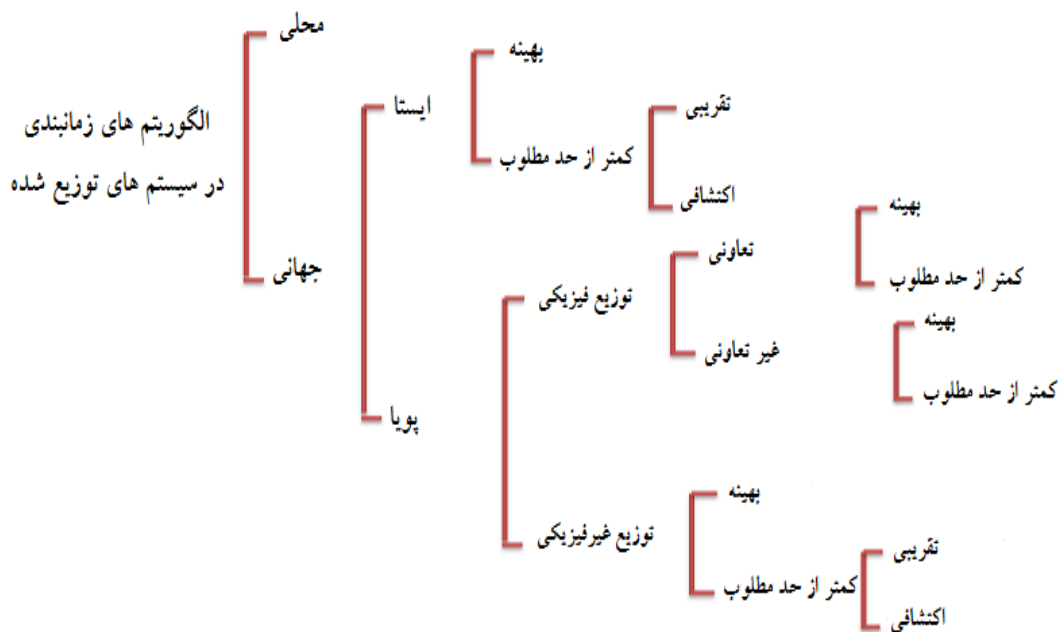
توازن بار حذف وظایف از ماشین‌های مجازی زیاد بارگذاری شده و تخصیص آن‌ها به ماشین‌های مجازی کم بارگذاری شده است. زمان‌بندی وظایف می‌تواند بر روی عملکرد کلی اجرای برنامه‌های یک سیستم تأثیر بگذارد. در الگوریتم‌های زمان‌بندی وظایف ایستا، تصمیمات مربوط به زمان‌بندی وظایف در زمان کامپایل ساخته خواهد شد، زمانی که منابع مورد نیاز برآورد می‌شوند. مزیت استفاده از این الگوریتم سادگی از نظر پیاده‌سازی و سربار است چون هیچ نیازی به نظارت مداوم آمار عملکرد گره‌ها وجود ندارد. الگوریتم‌های ایستا تنها زمانی که تنوع بار در ماشین‌های مجازی کم است به درستی کار می‌کنند. بنابراین، این الگوریتم‌ها برای محیط‌های محاسبات ابری موبایل که در آن تنوع بار در زمان‌های مختلف تغییر می‌کنند مناسب نیست. در

¹ Resource Discovery

² Resource Selection

³ Job Execution

الگوریتم‌های زمان‌بندی وظایف پویا، الگوریتم زمان‌بندی وظایف پویا برای توزیع بار کاری در میان گره‌ها در زمان اجرا تغییر ایجاد می‌کنند؛ آن‌ها اطلاعات فعلی بار را در زمان تصمیم‌گیری توزیع استفاده می‌کنند [۸]. به دلیل تنوع روش‌های ارائه‌شده برای مسئله زمان‌بندی، مقایسه کمی و کیفی این روش‌ها بسیار مشکل می‌باشد. قبل از بررسی یا ارائه یک الگوریتم زمان‌بندی، بهتر است جایگاه آن در مقایسه با سایر الگوریتم‌ها مشخص شود. با توجه به این که سیستم توزیع شده نوع خاصی از سیستم‌ها می‌باشد، الگوریتم‌های زمان‌بندی در گرید در زیرمجموعه‌ای از این طبقه‌بندی (شکل ۱) قرار می‌گیرند [۹].



شکل ۱: طبقه‌بندی الگوریتم‌های زمان‌بندی در گرید

در بالاترین سطح الگوریتم‌های زمان‌بندی به دو دسته سراسری^۱ و محلی^۲ تقسیم‌بندی می‌شوند. در روش زمان‌بندی محلی در مورد نحوه تخصیص فرآیندها به یک پردازنده و اجرای آن تصمیم‌گیری می‌شود. در روش سراسری با استفاده از اطلاعات سیستم‌ها تخصیص فرآیندها به چندین پردازنده برای بهینه‌سازی کارایی کلی سیستم صورت می‌گیرد. زمان‌بندی توزیع شده در شاخه زمان‌بندی سراسری قرار می‌گیرد [۱۰]. در سطح بعدی در سلسله‌مراتب زمان‌بندی، زمان‌بندی ایستا و پویا است. در الگوریتم‌های ایستا اطلاعات لازم درباره همه منابع موجود در محیط محاسباتی ناهمگن در لحظه زمان‌بندی موجود است. از مزایای این روش سادگی برنامه‌نویسی از دیدگاه زمان‌بند است. در مقابل زمان‌بندی پویا ایده اصلی تخصیص درخواست در زمان اجرای برنامه‌ها است. این روش در مواقعی مورد استفاده قرار می‌گیرد که تخمین هزینه برنامه‌ها یا درخواست‌های وارد شده به سیستم به صورت پویا مشکل است [۱۰].

^۱ Global

^۲ Local

زمان‌بندی درخواست‌ها به صورت پویا شامل دو قسمت اصلی است: تخمین حالت سیستم (علاوه بر روش تخمین هزینه در روش ایستا) و تصمیم‌گیری. تخمین حالت سیستم شامل جمع‌آوری اطلاعات از سراسر سیستم محاسباتی ناهمگن و سپس تخمین زدن می‌باشد. بر اساس تخمین به عمل آمده تصمیم انتساب یک درخواست به منبع انتخاب انجام می‌شود [۱۰]. هر دو نوع زمان‌بندی ایستا و پویا به طور گسترده در محاسبات توزیع شده پذیرفته می‌شوند.

الگوریتم‌های زمان‌بندی از دیدگاه دیگر به دو دسته بلادرنگ (آنلاین) و دسته‌ای تقسیم‌بندی می‌شوند. در حالت بلادرنگ به محض ورود کار عمل تخصیص کار به یک پردازنده در دسترس توسط زمان‌بند انجام می‌پذیرد در صورتی که در حالت دسته‌ای عمل تخصیص کار در طی زمان مشخصی که رخداد زمان‌بندی نامیده می‌شود، انجام می‌شود [۱۰].

دسته دیگر به دو دسته زمان‌بندی بهینه و تقریباً بهینه تقسیم‌بندی می‌شوند. در زمان‌بندی بهینه اختصاص بهینه می‌تواند بر اساس برخی توابع معیار مانند Makespan و استفاده بیشینه از منابع باشد. با توجه به پیچیدگی‌های موجود در محیط‌های محاسباتی ناهمگن بایستی بهینگی الگوریتم ثابت شود [۱۰].

دسته‌بندی دیگر در مورد الگوریتم‌های توزیع یافته در مقابل الگوریتم‌های متمرکز است. در سناریوهای زمان‌بندی پویا، مسئولیت اتخاذ تصمیمات زمان‌بندی سراسری می‌تواند به عهده زمان‌بند متمرکز باشد یا به وسیله زمان‌بندهای توزیعی چندگانه به اشتراک گذاشته شود. در محاسبات توزیع شده ممکن است برنامه‌های کاربردی زیادی ارائه شوند که لازم است به طور هم‌زمان دوباره زمان‌بندی شوند. روش متمرکز، مزیت سهولت اجرا را دارد اما مشکل نبود مقیاس‌پذیری، عدم تحمل خطا، امکان تبدیل شدن به گلوگاه را دارد [۱۱].

دسته‌بندی دیگر الگوریتم‌های تقریبی در مقابل الگوریتم‌های اکتشافی هستند. الگوریتم‌های تقریبی از مدل‌های محاسباتی رسمی استفاده می‌کنند اما به جای جستجوی کل فضای راه‌حل برای یک راه‌حل بهینه، به یک راه‌حل به اندازه کافی خوب راضی هستند. در این مورد، در شرایطی که یک معیار برای ارزیابی یک راه‌حل در دسترس باشد، این روش می‌تواند تا موقعی که زمان لازم برای یافتن یک زمان‌بندی مورد قبول کاهش یابد، به کار برده شود [۱۱].

در نهایت دسته‌بندی آخر مربوط به الگوریتم‌های زمان‌بندی تعاونی و غیرتعاونی است. هرگاه یک الگوریتم به عنوان الگوریتم زمان‌بندی توزیعی پذیرفته می‌شود، موضوع بعدی که باید در نظر گرفته شود این است که آیا گره‌هایی درگیر زمان‌بندی کار می‌باشند که به طور تعاونی کار می‌کنند یا مستقلانه در حالت کار غیرتعاونی، زمان‌بندهای اختصاصی تنها مثل هویت‌های خودمختار عمل می‌کنند و در تصمیمات با در نظرگیری موضوعات بهینه‌شان مستقل از تأثیرات تصمیم روی بقیه سیستم عمل می‌کنند.

روش‌های زیادی برای انتخاب بهترین داده میزبان و محاسبه میزبان وجود دارد. به عنوان مثال، می‌توانی با توجه به نزدیک بودن داده میزبان از نظر فاصله شبکه‌ای با محاسبه میزبان و یا با تلاش برای به حداکثر رساندن همکاری ارتباط بین دو داده میزبان، در مورد چگونگی اولویت دسترسی به برخی از مجموعه فایل‌های مشابه تصمیم‌گیری کرد [۱۲]. معیارهای زیادی برای طبقه‌بندی الگوریتم زمان‌بندی جریان کار وجود دارد که از جمله

این معیارها مدل زمان بندی، ماهیت الگوریتم زمان بندی، نوع الگوریتم زمان بندی، هدف بهینه سازی و محیط کاربردی است.

بر طبق تعداد اهداف بهینه سازی، مسئله زمان بندی جریان کار می تواند به عنوان مدل تک هدفی و چند هدفی حل شود. به عنوان مثال، الگوریتم اصلاح هوشمند قطرات آب مدل تک هدفی است، الگوریتم ژنتیک چند هدفی مدل چند برای کشف فضای جستجو و تولید مجموعه راه حل پارتو برای انتخاب است [۱۳].

جریان کار با بهینه سازی تابع های مختلف هدفمند تکمیل می شود، به طوری که سناریوهای مختلف کاربردی نیازمند الگوریتم متفاوت زمان بندی است. بر اساس اهداف بهینه سازی الگوریتم، می توان آن ها را بر حسب عملکرد محدود الگوریتم زمان بندی، معیارهای کیفیت سرویس کاربر و یا ترکیبی از هر دو طبقه بندی کرد [۱۴]. عملکرد محدود الگوریتم زمان بندی، تمرکز بر بهره وری انرژی، حفظ تعادل بار و در دسترس بودن و غیره دارد. معیار کیفیت سرویس کاربر محدود معیارهایی هستند که الگوریتم های زمان بندی را از دیدگاه کاربر و از لحاظ هزینه، زمان و قابلیت اطمینان بهینه می کنند؛ مانند الگوریتم ژنتیک مبتنی بر کیفیت سرویس و آگاهی از ترکیب خدمات و الگوریتم های بهینه زمان بندی ابرهای هیبرید.

الگوریتم های زمان بندی می تواند از نظر نوع الگوریتم، به اکتشافی و فرا اکتشافی تقسیم شوند. به عنوان مثال، الگوریتم اکتشافی چند هدفی و الگوریتم زمان بندی با مهلت محدود جزء الگوریتم های اکتشافی است. برای الگوریتم های فرا اکتشافی می توان الگوریتم اکتشافی مبتنی بر کیفیت سرویس و آگاهی از ترکیب خدمات را نام برد [۱۵]. هدف از الگوریتم های اکتشافی، ارائه راه حل در چارچوب یک زمان قابل قبول است که برای حل مسئله مناسب باشد، ممکن است الگوریتم اکتشافی، بهترین راه حل واقعی برای حل مسئله نبوده ولی می تواند راه حل نزدیک به بهترین باشد. الگوریتم های اکتشافی با الگوریتم های بهینه سازی برای اصلاح کارایی الگوریتم می توانند ترکیب شوند. الگوریتم فرا اکتشافی ترکیبی است از الگوریتم های اکتشافی که برای پیدا کردن، تولید یا انتخاب هر اکتشاف در هر مرحله طراحی می شود و راه حل خوبی برای مسائلی که مشکل بهینه سازی دارند ارائه دهد. الگوریتم های فرا اکتشافی برخی از فرضیات مسائلی بهینه سازی که باید حل شود را در نظر می گیرد ولی زمان طولانی لازم دارد [۱۶].

الگوریتم اولین زمان انجام ناهمگن سه فاز دارد (۱) فاز وزنی: تنظیم وزن ها در گره های و لبه های برنامه (۲) رده بندی فاز: ایجاد فهرست در دسته های سازماندهی شده (۳) فاز ترسیم کارها را در منابع تنظیم می کند. در مرحله وزنی، وزن ها در گروه لبه ها تنظیم می شود. وزن ها در گروه هایی تنظیم می شود که طبق زمان اجرای پیش بینی شده کارها، محاسبه می شود و وزن ها در لبه هایی تنظیم می شود که طبق زمان پیش بینی تنظیم شود که در آن داده منابع جابجا شده است [۱۷].

فاز رده بندی پیمایش رو به بالا انجام می شود و مقدار رده هر یک از کارها را تنظیم می کند. مقدار رده مساوی هم وزن گره بعلاوه زمان اجرای جانشین ها است. زمان اجرای جانشین برای هر لبه با جانشین آن گره برآورد می شود و وزن آن را در مقدار رده گره جانشین می افزاید و ماکسیمم خلاصه را انتخاب می کند. در فاز ترسیم، کارهای متوالی لیست رده بندی در منابع ترسیم می شود. برای هر کار، منابع، اولین زمان مدنظر را برای تکمیل اجرا انتخاب می کنند. در کل این الگوریتم در ابتدا میانگین زمان اجرای هر کار و میانگین زمان ارتباط منابع را

با دو کار موفق محاسبه می‌کند. سپس گردش کار برحسب انتخاب منبع، کارها را به ترتیب اولویت زمان‌بندی می‌کند و هر کار در منبعی تنظیم می‌شود که کار را در اولین زمان تکمیل می‌کند.

الگوریتم جدید زمان‌بندی که ترکیبی از هزینه و زمان می‌باشد و یک مدل چند هدفی که در محیط ابرهیب‌رید استفاده می‌شود. این الگوریتم زمان اجرا و هزینه را با ورودی کاربر در نظر می‌گیرد و روی حداقل رسانی هزینه‌ای متمرکز است که طبق آن کار موعده مقرر را تعریف می‌کند. عملکرد این الگوریتم به روش Min-Min آنالیز می‌شود [۱۸].

الگوریتم ژنتیک روشی است که خدماتی با بهترین مقادیر کیفیت سرویس ارائه می‌دهد. الگوریتم ژنتیک چهار مرحله: انتخاب، عبور، جهش و بررسی دارد و این الگوریتم جزء الگوریتم‌های چند هدفی می‌باشد. این الگوریتم هزینه جریان کار و معیارهای ساخت را در محیط محاسبات ابری موبایل به حداقل می‌رساند. این الگوریتم محدودیت هزینه‌ای دارد و بیشتر محدودیت کیفیت سرویس در الگوریتم زمان‌بندی در محدودیت هزینه است [۱۹].

الگوریتم زمان‌بندی هزینه بهینه ابری^۱ برای ارتباط با زمان‌بندی ابرهای هیبریدی بکار می‌رود. این الگوریتم تصمیم می‌گیرد که منابع باید از ابر عمومی سست شود و به ابر خصوصی برای ارائه توان کافی پردازش و اجرای کار در زمان اجرا داده بیوندد. محققین آزمایشاتی انجام دادند که نشان می‌دهد الگوریتم زمان‌بندی هزینه بهینه ابری هزینه را کاهش می‌دهد در حالی که زمان اجرای مدنظر را کسب می‌کند [۲۰].

یکی از جنبه‌های بسیار مهم برای سیستم ابری مدل تجارت بازار محور است که چالش‌های زیادی برای استراتژی‌های زمان‌بندی برنامه کار را به همراه دارد. استراتژی زمان‌بندی سلسله‌ای بازار محور برای سیستم برنامه ابری چند هدفی پیشنهاد شد. در این زمان‌بندی سطح خدمات با تنظیمات کار با سرویس و زمان‌بندی سطح کار با بهینه‌سازی تنظیم کار با ماشین‌های مجازی مراکز ابری سروکار دارد [۲۱].

الگوریتم انشعاب جریان کار تطبیقی^۲ (AWS) به این صورت عمل می‌کند. زمانی که زمان‌بندی کار را دریافت می‌کند، ابتدا آن DAGG کار را به منطق منتقل می‌کند تا CCR آن و تعداد دسته‌های منبع را محاسبه کند، که در آن CCR میانگین نسبی هزینه محاسبه در هزینه ارتباط است. مقدار بالای CCR نمودار کار محاسبه جامع است. پس، زمان‌بندی دسته‌های منبع N را انتخاب می‌کند که بالاترین رده N را مطابق شاخص دارد. تکرار درصد رشد نموداری هر گره را در G چک می‌کند تا بررسی کند آیا گره می‌تواند در زیر نمودار G قرار گیرد که با بررسی بزرگ‌ترین لبه ارتباطی شروع می‌شود که دو گره آن چک نشده است. اگر هیچ لبه‌ای نباشد به این معنی است که کل گره‌های چک نشده در G همدیگر ایزوله شده است، این کارها در صورت ممکن در زیر نمودار ظاهر می‌شود.

هدف پیمایش نمودار، بروز رسانی بیشتر کارها در زیر نمودار فعلی است و محدودیت آستانه منبع را مختل نمی‌کند و در عین حال، کارها غیرضروری را دنبال می‌کند و نمی‌تواند توازن را افزایش دهد ولی هزینه ارتباط

^۱ Hybrid Cloud Optimized Cost (HCOC)

^۲ Adaptive Workflow Splitting

درون دسته را افزایش می‌دهد [۲۲]. برای دست‌کاری گره کار در پیمایش از پشته استفاده می‌شود. اگر این لبه نمی‌تواند در G محدود آشکار شود، گره مسئول مشخص می‌شود به این معنی که زیر نمودار جدید درست می‌شود. زمان‌بندی حلقه گره‌ها را در G تکرار می‌کند و در دسته‌های منبع تنظیم می‌کند و الگوریتم پایان می‌یابد، زمان‌بندی هر زیر نمودار را در دسته منبع تنظیم شده‌اش گزارش می‌کند.

کارهای قبلی که در زمینه‌ی زمان‌بندی ماشین‌های مجازی و پارتیشن‌بندی کد وجود دارند دارای اهداف مختلفی می‌باشند. برخی از الگوریتم‌ها درصدد انتخاب میزبان مناسب برای درخواست ماشین‌های مجازی با هدف کاهش مصرف انرژی در مراکز داده می‌باشند. برخی از کارهای قبلی از راه‌حل‌های بهینه‌سازی چندمنظوره برای حل مسئله‌ی زمان‌بندی وظایف و پارتیشن‌بندی کد استفاده کرده‌اند. در برخی دیگر از کارهای تکمیل‌شده، هدف کاهش هزینه برای کاربران درخواست دهنده‌ی ماشین مجازی می‌باشد. در این روش‌ها فراهم‌کننده‌ی سرویس تعدادی ماشین مجازی را از ابتدا برای خود رزرو می‌کند و در صورت نیاز ماشین اضافی را درخواست می‌دهد. تابع هدف برای زمان‌بندی ماشین مجازی و پارتیشن‌بندی کد در این روش بر مبنای کاهش هزینه می‌باشد. همچنین برخی از کارها باهدف کاهش تعداد ماشین‌های فیزیکی فعال سعی بر کاهش مصرف انرژی و هزینه دارند و این‌گونه الگوریتم‌ها با افزایش تعداد درخواست‌ها کارایی چندانی ندارند. در سال‌های اخیر برخی از الگوریتم‌ها باهدف کاهش ترافیک موجود بر روی ماشین‌های فیزیکی روش پویایی را برای زمان‌بندی ماشین‌های مجازی ارائه می‌دهند و این روش‌ها با کمترین نیاز به مهاجرت روش بهینه‌ای برای زمان‌بندی وظایف و پارتیشن‌بندی کد ماشین‌های مجازی می‌باشند [۲].

اغلب روش‌های قبلی که برای جایابی ماشین مجازی بیان‌شده است با هدف تأمین و بهبود کارایی یکی از معیارهای کیفیت سرویس می‌باشد. در این پژوهش روشی را ارائه می‌دهیم که مسئله‌ی زمان‌بندی وظایف و پارتیشن‌بندی کد را باهدف افزایش توازن بار که یکی از مهم‌ترین معیارهای کیفیت سرویس می‌باشد در نظر می‌گیرد. الگوریتم پیشنهادی به ما کمک می‌کند تا با استفاده حداکثر از منابع سبب توازن مناسب بار در مراکز داده‌ای ابری سیار شویم و درعین حال با در نظر گرفتن قابلیت‌های هر یک از ماشین‌های فیزیکی و استفاده از روش مناسب جهت توزیع درخواست‌های ماشین علاوه بر استفاده‌ی بهینه از منابع سبب افزایش توازن بار و تأمین معیارهای کیفیت سرویس شویم.

روش پیشنهادی

تحقیقات بسیاری وجود دارند که به طور کامل در حوزه‌ی الگوریتم‌های زمان‌بندی انجام شده‌اند. اکثر آن‌ها بر تعادل بار برای سرورهای وب سنتی تمرکز می‌کنند. یکی از مشکلات چالش‌برانگیز برنامه‌ریزی در مراکز داده ابری در نظر گرفتن هر دو عملیات تخصیص و مهاجرت (انتقال) ماشین‌های مجازی با قابلیت پیکربندی مجدد و ویژگی‌های یکپارچه‌ی ماشین‌های فیزیکی در حال میزبانی است. برخلاف الگوریتم‌های برنامه‌ریزی برقراری توازن بار سنتی که تنها سرورهای فیزیکی با در نظر گرفتن معیارهای کارایی متفاوت و با مدنظر قرار دادن عواملی چون بار پردازنده، حافظه و پهنای باند شبکه یکپارچه برای هر دو ماشین فیزیکی و ماشین‌های مجازی و در این پژوهش معیار یکپارچه‌ای برای برقراری توازن و تعادل بار بر روی هر سرور ایجاد می‌شود. در این

پژوهش یک الگوریتم زمان‌بندی یکپارچه و پویای منابع بر اساس الگوریتم تکاملی بهینه‌سازی ازدحام ذرات معرفی خواهد شد که پردازنده، حافظه و پهنای باند یکپارچه شده برای هر دو ماشین فیزیکی و ماشین‌های مجازی را اصلاح می‌کند.

در محیط ابری، معماری تخصیص وظیفه یک مدل سه لایه است، یعنی، درخواست وظیفه و کار، مدیریت منابع و اجرای وظیفه. در لایه اول، کاربران با سیستم تعامل می‌کنند؛ در لایه میانی، مدیریت منبع یک نقش قابل توجه و برجسته در تقسیم کار به وظایف، تخصیص وظایف و مدیریت منابع بازی می‌کند؛ در آخرین لایه وظایف به عنوان توالی‌های از پیش تعریف شده اجرا می‌شوند. مدیریت مراکز داده‌ای کلیدی برای تخصیص وظایف است.

هدف اصلی جایابی، به حداکثر رساندن استفاده از منابع، به حداقل رسانی زمان اجرای ماشین‌های مجازی و با افزایش توازن بار در محیط محاسباتی ابری پس از رفع محدودیت‌های ماشین‌های مجازی و منابع سیار می‌باشد. ما به دنبال برقراری تعادلی مناسب بین سطح موردنیاز کیفیت خدمات می‌باشیم. در این پژوهش، یک روش زمان‌بندی و پارتیشن‌بندی کد با حالت پویای مبتنی بر تکامل چند هدفی در شبکه را پیشنهاد داده‌ایم. رویکرد جایابی ما از روش بهینه‌سازی چند هدفی با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات برای پیدا کردن جایابی‌های کارآمد استفاده کرده است. این روش دست به کاوش فضای جستجو به‌طور روشن زده تا از ایستایی جلوگیری کرده و راه‌حل نزدیک به بهینه را تولید نماید.

پیدا کردن راه‌حل نزدیک به بهینه برای مشکل زمان‌بندی وظایف و پارتیشن‌بندی کدمنعطف با بیش از یک هدف در یک کارآمد از نظر زمانی، کاری دشوار است. محیط ابر دارای ماهیت پویا بوده و تخصیص دوباره ماشین‌های مجازی در آن کاملاً آشکارکننده این وضعیت می‌باشد. هدف این پژوهش، ارائه یک جایابی بهتر به مجریان ابر بوده که دارای وضعیت بهتری از نظر توازن بار و کیفیت خدمات خواهد بود. این جایابی‌کننده می‌تواند از عهده رفتار پویای منابع، محدودیت‌های منبع و محدودیت‌های مربوط به ماشین‌های مجازی پیشین برآید.

روش جایابی ارائه‌شده در این پژوهش، فراهم‌کننده مزیت بیشتری در رابطه با اکثر ویژگی‌های ابر چه از نظر سرویس دهنده و چه از نظر مشتریان می‌باشد. محیط پویای شبکه روش پیشنهادی را تبدیل به یک روش جایابی پویا با زمان واقعی می‌سازد. یک روش گروه‌بندی ماشین‌های مجازی برای گروه‌بندی ماشین‌های مجازی به‌خوبی شبکه‌ای شده به‌منظور تسهیل در محاسبات پیچیده پیشنهاد گردیده است. انجام آزمایشات بر روی مجموعه‌های مختلف داده‌ها و پارامترهای گوناگون نشان‌دهنده کارآمدی معیارهای جایابی چند هدفی و استخراج عملکرد از منبع شبکه بوده است. رویکرد ما در رابطه با حل مشکلات بالا با استفاده از الگوریتم تکاملی برای کوچک کردن حیطه اهداف چندگانه به‌اندازه‌ای مناسب بوده که راهبرد جایابی نزدیک به بهینه را به‌موقع پیدا کند. الگوریتم بهینه‌سازی ازدحام ذرات به‌عنوان یک چهارچوب بنیادین برای ماژول زمان‌بندی

وظایف و پارتیشن‌بندی کد ما بکار گرفته شده است. ثابت شده است که زمان‌بندی وظایف و پارتیشن‌بندی کد یک مسئله چندجمله‌ای غیرقطعی می‌باشد.

روش جایابی مبتنی بر الگوریتم بهینه‌سازی ازدحام ذرات می‌تواند به‌عنوان یک روش جایابی زمان واقعی به دلیل افزایش در قابلیت محاسباتی پردازشگرها در پنج سال گذشته، عمل کند. این جایابی یک روش جستجو بوده که در آن شایستگی به صورتی پویا بر مبنای واریانس‌های مقادیر تناسب در هر نسل تغییر می‌کنند. این روش جایابی، به حداقل رسانی زمان تکمیل ماشین‌های مجازی نیز توجه می‌کند. ما در این پژوهش بر مبنای الگوریتم تکاملی چند هدفی، مسئله جایابی منبع در شبکه را به مسئله جایابی پروژه محدود از نظر منبع تبدیل کرده‌ایم. در اینجا ما از سازوکار جایابی پویا، عامل پیشرفته یادگیری که به دنبال بهینه‌سازی پنج هدف می‌باشد، استفاده کرده‌ایم. ماژول چند هدفی زمان‌بندی وظایف و پارتیشن‌بندی کد با راهبرد گروه‌بندی ماشین‌های مجازی مبتنی بر نمودار برای ماشین‌های مجازی به‌خوبی دانه‌دانه شده، این فرآیند را آسان می‌کند.

به منظور ارزیابی جنبه‌های مثبت و منفی رابطه نقشه‌برداری از کارگران M با وظایف $TaskNum$ ، باید کل زمان مصرفی یک مدل نقشه‌برداری معین کار شود و مقایسه متقابل انجام شود. به منظور آنالیز کل زمان اجرا وظایف ارائه شده، در ابتدا تعداد وظایف در ماشین مجازی t به عنوان $TaskNum(t)$ به طور کمی تعیین می‌شود که توسط فرمول یک مشخص می‌شود که J تعداد کل وظایف است، N مخفف شماره گماشین‌های فیزیکی در مراکز داده‌ای ابری است.

$$taskNum(t) = \sum_{t=1}^J TaskNum(t) \quad (1)$$

برای تمایز بین این وظایف، وظیفه‌ی j^{th} در وظیفه i^{th} با m با استفاده از فرمول (۲) شناسایی می‌شود،

$$m = \sum_{k=1}^{i-1} TaskNum(k) + j \quad (2)$$

ماتریس ETC ($tasknum \times workernum$) (زمان مورد انتظار برای محاسبه) با توصیف زمان مصرفی وظیفه i^{th} انجام شده با محاسبه کننده j^{th} را معرفی می‌کند که $tasknum$ به شماره وظایف اشاره دارد و $workernum$ تعداد واحدهای محاسباتی را بیان می‌کند. بنابراین با توجه به رابطه رمزگشایی شده بین پردازشگرها و وظایف و ماتریس ETC، زمان اجرای کار t^{th} ایجاد می‌شود.

$$JopTime(t) = \max_{i=1}^{TaskNum(t)} \sum_{j=1}^k TaskTime(j, i) + j \quad (3)$$

که نشان دهنده‌ی وظیفه‌ی i^{th} در موقعیت تخصیص داده شده به ماشین مجازی t است، $\text{TaskTime}(j,i)$ نشان دهنده‌ی هزینه اجرای وظیفه i در ماشین فیزیکی j است. میانگین زمان اجرا یک ماشین مجازی توسط فرمول (۴) مشخص می‌شود.

$$\text{avgJobTime} = \frac{\sum_{t=1}^J \text{JobTime}(t)}{J} \quad (۴)$$

محدوده و ظرفیت وظیفه شامل سه بخش است، یعنی، زمان محاسبه وظیفه، دیسک I/O زمان و انتقال داده‌ها از راه دور. برای یک وظیفه i ، $1 \leq i \leq \text{taskNum}(t)$ نشان دهنده‌ی زمان انجام وظیفه i^{th} است و چنین مشخص می‌شود.

$$T_i = T_{\text{computing}_t} + T_{\text{diskI/O}_t} + T_{\text{transmission}_t} \quad (۵)$$

$T_{\text{computing}}$ هنگامی که با دیگر هزینه‌های زمان در خوشه‌ها محاسبه شود، نسبتاً کوتاه است. دیسک I/O زمان مورد نیاز برای انتقال داده‌ها از دیسک محلی را نشان می‌دهد که با محیط سخت‌افزار خاصی مشخص می‌گردد. $T_{\text{transmission}}$ هزینه زمانی انتقال داده‌ها از راه دور است. در این پژوهش، تأخیر زمانی شبکه محاسبه نمی‌شود، از ماتریس $\text{DTC} (\text{taskNum} \times N)$ برای نشان دادن هزینه انتقال داده‌ها استفاده می‌شود که $\text{DTC}[i,j]$ هزینه وظیفه‌ی برنامه‌ریزی شده i^{th} برای ماشین فیزیکی j^{th} است، اگر یک تکرار داده در گره j^{th} وجود داشته باشد، $\text{DTC}[i,j]=0$. به طور پیش فرض سه استراتژی تکرار داده در سیستم توزیع شده اتخاذ می‌شود. به طوری که کل زمان مصرفی وظایف ماشین مجازی به صورت رابطه (۶) است:

$$\text{TotalTime} = \max_{w=1}^N \sum_{i=1}^n \text{node}(w,i) \quad (۶)$$

که گره (w,i) به هزینه زمانی وظیفه i^{th} اجرا شده در ماشین مجازی w^{th} اشاره دارد که شامل محاسبه، دیسک I/O و انتقال داده از راه دور است. و n تعداد کل وظایف در این گره است.

میانگین بهره‌وری پردازنده در یک سرور منفرد i با CPU^U_i نشان داده می‌شود. برای به دست آوردن این معیار از بهره‌وری پردازنده در طی دوره‌ی مشاهده میانگین گرفته می‌شود. برای محاسبه برای مثال، اگر دوره مشاهده یک دقیقه و کاربرد پردازنده هر ۱۰ ثانیه ثبت شود، پس CPU^U_i میانگین شش مقدار ثبت شده از سرور i است.

CPU^n_i تعداد کل CPU‌های سرور i باشد، آنگاه میانگین کاربرد همه CPU‌ها در یک مرکز داده‌ی ابری به صورت رابطه ۷ محاسبه می‌شود:

$$CPU_u^A = \frac{\sum_i^N (CPU_i^U) CPU_i^n}{\sum_i^N CPU_i^n} \quad (7)$$

که N تعداد کل سرورهای فیزیکی در یک مرکز داده‌ی ابری است. به طور مشابه، میانگین کاربرد حافظه، پهنای باند شبکه سرور i ، همه حافظه‌ها و همه پهنای باند شبکه در یک مرکز داده ابری می‌تواند به ترتیب به صورت $MEM_u^A, NET_u^A, MEM_u^U, NET_u^U$ تعریف شود.

در این تحقیق، الگوریتم بهینه‌سازی ازدحام ذرات چندهدفه روی مسئله زمان‌بندی کارهای غیر وابسته ارائه شده است. در این تحقیق از یک مدل ذره خاص استفاده شده است که یک راهبرد زمان‌بندی وظایف و پارتیشن‌بندی کد بر روی مراکز داده‌ای ابری که برآورده‌کننده محدودیت‌های مختلف اجرایی بوده، توسط ذره ارائه شده است.

الگوریتم‌های بهینه‌سازی ازدحام ذرات کلاسی از الگوریتم‌های جستجوی بهینه هستند که تکامل بیولوژیکی و مکانیسم هوش ازدحامی را شبیه‌سازی می‌کنند. بعد از اینکه جمعیت اولیه تولید می‌شود، راه‌حل‌های تقریبی بهتر و بهتری بر اساس سازگاری از نسلی به نسل دیگر تکامل می‌یابد. در طی هر نسل، فرد بر اساس سازگاری افراد مختلف در دامنه‌ی مشکل خاصی انتخاب می‌شود. بر اساس وضعیت واقعی محاسبات ابری. این پژوهش استراتژی زمان‌بندی و زمان‌بندی وظایف و پارتیشن‌بندی کد از طریق الگوریتم بهینه‌سازی ازدحام ذرات بهبود یافته پیشنهاد می‌کند.

یک طرح شماتیک برنامه‌ریزی می‌تواند توسط ذره‌ای بیان شود که به شکل گره-وظیفه کد می‌شود [۲۳]. طول ذره دقیقاً برابر با تعداد ماشین‌های مجازی است. مقدار یک ژن برابر با تعداد ماشین‌های فیزیکی است که در آن وظیفه این ژن مطابقت است با اجرا. با توجه به ذره زیر،

{1,3,2,4,3,1,4,2,3,1,3,2,4,1}

اولین ماشین مجازی در مرکز داده‌ای ابری اول اجرا می‌شود، دومین ماشین مجازی در مرکز داده‌ای ابری سوم اجرا می‌شود. منظور از $Node_i$ بیانگر ماشین فیزیکی i ام می‌باشد که بیانگر ماشین‌های مجازی منتسب شده به آن می‌باشد.

$Node_1 : \{1,6,10\}, Node_2 : \{3,8,12\},$

$Node_3 : \{2,5,9,11\}, Node_4 : \{4,7,13\},$

الگوریتم‌های ازدحام ذرات ساده به صورت تصادفی جمعیت اولیه را ایجاد می‌کنند و افراد عملکرد پایینی دارند. در این پژوهش، الگوریتم حریم‌صانه^۱ استفاده می‌شود. جمعیت اولیه از طریق انجام انتخاب بهینه بر اساس وضعیت فعلی ایجاد می‌شود.

¹¹ Greedy algorithm

توازن بار عبارت است از توزیع درخواست‌ها میان چندین ماشین مجازی است که همگی سرویس یکسانی را برای کاربران ارائه می‌دهند. فرایند توازن بار به صورت شفاف انجام می‌گیرد و به این معنی است که فرایند از دید کاربران مخفی صورت می‌گیرد و کاربران از چگونگی توزیع بار و اینکه از کدام ماشین مجازی واقع در ابر پاسخ می‌گیرند هیچ اطلاعی ندارند. در این پژوهش یک تابع سازگاری بر اساس چندین هدف کارایی مختلف تعریف می‌شود. اهداف کارایی شامل موارد زیر است:

(۱) زمان مصرفی کمتر (۲) واریانس کمتر برای بار مراکز داده‌ای و ماشین‌های فیزیکی (۳) زمان پاسخ (۴) میزان حافظه باقیمانده

- زمان مصرفی کمتر

برای محاسبه معیار کل از جمع زمان اجرای ماشین‌های مجازی اختصاص داده شده به یک ماشین فیزیکی و مراکز داده‌ای استفاده می‌گردد که این امر از طریق رابطه (۶) فراهم می‌گردد که با $totalTime_{CDC}$ نشان داده می‌شود.

- واریانس کمتر برای بار مراکز داده‌ای و ماشین‌های فیزیکی

برای محاسبه معیار دوم می‌توان اظهار نمود که حجم بار بر روی ماشین برابر با مجموعه فضای اشغال شده تراکنش‌های صف ورودی ماشین مجازی می‌باشد. و موجب افزایش مصرف حافظه در ماشین مجازی می‌شود.

- زمان پاسخ

زمان پاسخ از مهمترین پارامترهای مورد بحث در مورد سیستم‌ها می‌باشد. و سیستم‌های محاسبات ابری جدا از این ویژگی نمی‌باشند. این زمان عبارت است از زمان تحویل یک تراکنش به یک ماشین مجازی تا زمان اولین اجرای آن. زمان پاسخ مربوط به یک ماشین مجازی براساس رابطه ۲ محاسبه می‌شود. که در این رابطه PT_i مدت زمان پردازش یک تراکنش، r_i برابر با زمان دریافت پاسخ و s_i برابر با زمان ارسال تراکنش به ماشین مجازی می‌باشد.

$$WT = \frac{\sum_{i=1}^m r_i - s_i - PT_i}{m} \quad (8)$$

- میزان حافظه باقیمانده

برابر با مقدار فضای باقیمانده با وجود تمامی تراکنش‌های موجود در صف تراکنش‌های ماشین مجازی می‌باشد. این پارامتر به دلیل پیشگیری از مشکل کمبود حافظه دارای اهمیت زیادی می‌باشد. لذا از این رو ما این پارامتر را مد نظر قرار دادیم. این معیار با Mem_{ava} نشان داده می‌شود و به معنای میزان حافظه در دسترس می‌باشد.

روش پیشنهادی باید توازن مناسبی بین این اهداف برقرار سازد. همچنین از طرف دیگر این اهداف ممکن است بسته به کاربردهای مختلف دارای اولویت‌های متفاوتی باشند. بنابراین برای رسیدن به این هدف، باید به کاربر این اختیار را داد تا توازن مناسبی بین این فاکتورها برقرار سازد و برای این کار کاربر یک سری از وزن‌ها را به هر یک از این اهداف می‌دهد. بنابراین تابع شایستگی روش پیشنهادی، در رابطه ۹ نمایش داده شده است:

(۹)

$$\text{Fitness} = \alpha \times ILB + \beta \times \text{totalTime} + \varphi \times WT + \rho \times Mem_{ava}$$

که در آن α و β ضرایب مختلف وزنی می‌باشد که به هر یک از اهداف کارایی می‌باشد و در آن $\alpha + \beta + \varphi + \rho = 1$ می‌باشد.

مراحل اصلی الگوریتم جایابی پیشنهادی به صورت زیر می‌باشند:

مرحله ۱: به دست آوردن اطلاعات کاربرد دوره‌ی مشاهده‌ی فعلی، شامل [کاربرد CPU، کاربرد حافظه، کاربرد شبکه].

مرحله ۲: پذیرش درخواست وظیفه جدید (از بالاترین به پایین‌ترین اولویت صف‌ها به صورت صف‌های در حال انتظار، بهینه‌سازی، حذف می‌باشند).

مرحله ۳: بررسی صف در حال انتظار، اگر خالی نباشد، همان صف در حال انتظار، سپس مرحله ۷ و فرایند تخصیص خاص برداشته می‌شوند. اگر عمل تخصیص موفق نشود، پس به مرحله ۸ می‌رود. اگر عمل تخصیص موفق باشد، به مرحله ۱ باز می‌گردد.

مرحله ۴: بررسی صف درخواست، با توجه به زمان شروع وظیفه اگر خالی نباشد، پس تخصیص به دنبال الگوریتم تخصیص برای مرحله ۷ شروع می‌شود، اگر موفق‌آمیز باشد، درخواست در صف حذف قرار می‌گیرد.

مرحله ۵: بررسی صف بهینه‌سازی، اگر خالی نباشد به مرحله ۹ می‌رود.

مرحله ۶: بررسی صف حذف، با توجه به زمان اتمام وظیفه آن را حذف می‌کند.

مرحله ۷: اجرای الگوریتم جایابی بر اساس الگوریتم بهینه‌سازی ازدحام ذرات با تابع هدف پیشنهادی.

مرحله ۸: اگر همه سرورهای فیزیکی نتوانند تخصیص داده شوند، پس وظیفه به صف انتظار در مد اولین ورود-اولین - سرویس^۱ اضافه می‌شود.

مرحله ۹: فرایند مهاجرت (انتقال): با توجه به آستانه‌های کاربرد از پیش تعیین شده CPU، حافظه، پهنای باند، الگوریتم سرور فیزیکی را با بالاترین بار و ماشین مجازی با کمترین بار برای مهاجرت با استفاده از الگوریتم تخصیص را می‌یابد. مهاجرت تا زمانی که کاربرد سرور فیزیکی زیر آستانه است، یا تکراری از مهاجرت همان ماشین مجازی وجود دارد، ادامه دارد.

مرحله ۱۰: بعد از بررسی همه صف‌ها، به مرحله ۱ برگردد.

¹ First-come-first-service fashion

برای پیاده‌سازی روش پیشنهادی از نرم‌افزار شبیه‌ساز محیط محاسبات ابری CloudSim 3.0.3 در محیط زبان برنامه‌نویسی Java استفاده گردیده است. این شبیه‌سازی بر روی دستگاهی کامپیوتری با پردازنده corei7 با میزان حافظه اصلی ۸ گیگابایت و بر روی سیستم‌عامل ویندوز ۱۰ اجرا شده است.

برای شبیه‌سازی روش پیشنهادی از مدل استاندارد^۱ SDSC^۱ مربوط به پایگاه داده سن دیگو استفاده شده است. این مجموعه داده متشکل از ۷۳۴۹۶ وظیفه می‌باشد که مشخصات هر یک از این وظایف از قبل مشخص است. در این پژوهش از ویژگی‌های زمان ورود، زمان اختصاص پردازنده، زمان اجرای کل، میزان عملیات ورودی/خروجی استفاده می‌شود و بر مبنای این مشخصات عملیات زمان‌بندی ایجاد شده و فیلدهایی چون زمان اجرا، زمان پاسخ و زمان اجرای کل تنظیم می‌گردد و جهت ارزیابی روش پیشنهادی مورد استفاده قرار می‌گیرد. در شبیه‌سازی‌های بکار رفته تعداد وظایف بین ۲۰۰ تا ۲۰۰۰ در نظر گرفته شده است. تعداد ماشین‌های مجازی برابر با ۵۰ بوده که هر یک دارای یک پردازنده می‌باشد. تعداد مراکز داده‌ای و میزبان‌ها برابر با ۵۰ در نظر گرفته شده است. برای ارزیابی کارایی روش پیشنهادی از معیار زمان اجرای کل استفاده می‌شود که بیانگر بزرگ‌ترین زمان در بین زمان‌های خاتمه منابع محاسباتی می‌باشد. در واقع این معیار بیانگر زمانی است که می‌توان از اتمام تمامی وظایف محوله به مراکز داده‌ای اطمینان حاصل نمود.

بحث و ارزیابی

روش پیشنهادی بر پایه الگوریتم فراابتکاری بهینه‌سازی ازدحام ذرات چندهدفه با برخی از روش‌های موجود در زمینه زمان‌بندی فعالیت‌ها در محیط محاسبات ابری مورد ارزیابی قرار می‌گیرد. برای مقایسه از الگوریتم ارائه‌شده در [۲۳]، [۲۴] و [۲۵] استفاده شده است. دلیل استفاده از روش ارائه‌شده در [۲۳] این است که از یک الگوریتم تکاملی قهرمانی در لیگ (LCA) بهره می‌برد. برای اولین مقایسه زمان اجرای کل روش پیشنهادی در مقایسه با روش‌های پیشین بر اساس زمان اجرای کل بر حسب تغییر در تعداد وظایف مقایسه می‌گردد. روش ارائه شده در [۲۴] نیز از الگوریتم ژنتیک و در [۲۵] از روشی مبتنی بر الگوریتم بهینه‌سازی ازدحام ذرات استفاده می‌نماید. برای این کار تعداد وظایف همانند روش ارائه شده در [۲۳] از ۲۰۰ تا ۲۰۰۰ و با فاصله ۲۰۰ تایی قرار گرفته و در هر حالت زمان اجرای کل مورد محاسبه و ارزیابی قرار گرفته است. نتایج بدست آمده در جدول ۱ نشان داده شده اند:

^۱ The San Diego Supercomputer Center (SDSC)

جدول ۱: مقایسه زمان اجرای روش‌های مختلف

تعداد وظایف	زمان اجرای کل روش پیشنهادی	زمان اجرای کل روش ارائه شده در [۲۳]	زمان اجرای کل روش ارائه شده در [۲۴]	زمان اجرای کل روش ارائه شده در [۲۵]
۲۰۰	۳۵	۴۰	۴۳	۵۱
۴۰۰	۸۰	۹۰	۹۸	۱۰۲
۶۰۰	۱۵۰	۱۷۰	۱۸۵	۱۹۲
۸۰۰	۱۹۵	۲۳۰	۲۵۴	۲۶۵
۱۰۰۰	۲۸۵	۳۲۰	۳۶۱	۳۸۲
۱۲۰۰	۳۷۰	۴۴۰	۴۸۹	۵۰۳
۱۴۰۰	۵۴۰	۵۹۰	۶۲۳	۶۴۵
۱۶۰۰	۶۲۰	۶۸۰	۷۳۰	۷۵۱
۱۸۰۰	۷۳۵	۸۰۰	۸۴۰	۸۷۸
۲۰۰۰	۸۱۰	۸۸۰	۹۲۱	۹۵۳

همان‌طور که از جدول ۱ به خوبی مشخص است روش پیشنهادی این پژوهش بر اساس مقایسه با روش ارائه شده در مطالعات قبلی، کارایی مناسبی را از خود نشان داده است و در تمامی حالت‌های اجرایی مختلف بر حسب تعداد وظایف متفاوت دارای عملکرد مناسبی بوده و دارای زمان اجرای کمتری نسبت به سایر روش‌های موجود دارد. این امر به علت استفاده از الگوریتم بهینه‌سازی ازدحام ذرات چندهدفه با تابع هدف مناسب در این پژوهش است که به خوبی قادر به جستجوی فضای مسئله و اجرای مناسب عمل زمان‌بندی می‌باشد. نمودار ارائه شده در شکل ۲ نیز این مقایسه را به خوبی نشان می‌دهد.



شکل ۲: نمودار مقایسه زمان اجرای روش پیشنهادی و سایر روش‌ها

روش پیشنهادی با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات چندهدفه، بارکاری را با انتقال پویای حجم کار محلی از یک ماشین به ماشین دیگر در گره راه دور و یا ماشینی که کمتر مورد استفاده قرار می‌گیرد، موازنه می‌کند و از این طریق زمان اجرای کل را به خوبی کاهش داده است. این عمل رضایت کاربر را حداکثر، زمان

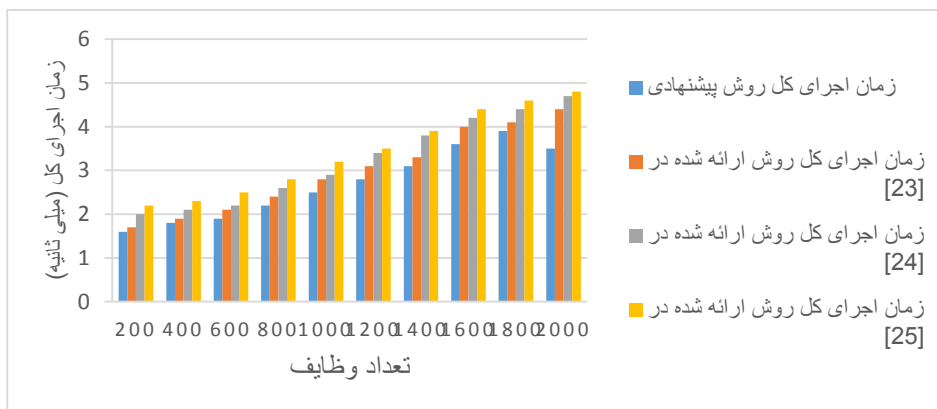
پاسخ را حداقل، بهره‌برداری از منابع را افزایش، تعداد رد کارها را کاهش و کار آیی سیستم را بالا می‌برد که در این بخش زمان اجرای کل با سایر روش‌ها مورد مقایسه قرار گرفت و در ادامه سایر اهداف کارایی مورد بررسی قرار می‌گیرد. این الگوریتم نه تنها تعادل بار را انجام می‌دهد، بلکه زمان پاسخ وظایف نگاشت شده به ماشین‌های مجازی سربارگذاری شده را نیز در نظر می‌گیرد و زمان‌بندی و توازن بار را به خوبی انجام می‌دهد و انتظار می‌رود که به خوبی قادر به توزیع متوازن وظایف باشد.

زمان پاسخ بیانگر فاصله زمانی بین درخواست اجرا و شروع اجرا توسط پردازنده‌های موجود در مراکز داده‌ای ابری میزبان می‌باشد. این معیار به یک معیار استاندارد در مطالعات موجود در حوزه زمان‌بندی تبدیل شده است. در این بخش مقایسه‌ای بین روش پیشنهادی و مطالعات انجام شده قبلی بر حسب زمان پاسخ ارائه می‌گردد. در واقع این معیار بیان‌گر نسبت عدالت از طریق پاسخگویی در استفاده مناسب از منابع می‌باشد. برای مقایسه روش پیشنهادی با مطالعه پیشین، حالتی بر حسب تعداد وظایف مختلف در نظر گرفته می‌شود (جدول ۲) نتایج این مقایسه‌ها را به خوبی نشان می‌دهند.

جدول ۲: مقایسه زمان پاسخ روش‌های مختلف

تعداد وظایف	زمان پاسخ روش پیشنهادی	زمان پاسخ روش ارائه شده در [۲۳]	زمان پاسخ روش ارائه شده در [۲۴]	زمان پاسخ روش ارائه شده در [۲۵]
۲۰۰	۱,۶	۱,۷	۲,۰	۲,۲
۴۰۰	۱,۸	۱,۹	۲,۱	۲,۳
۶۰۰	۱,۹	۲,۱	۲,۲	۲,۵
۸۰۰	۲,۲	۲,۴	۲,۶	۲,۸
۱۰۰۰	۲,۵	۲,۸	۲,۹	۳,۲
۱۲۰۰	۲,۸	۳,۱	۳,۴	۳,۵
۱۴۰۰	۳,۱	۳,۳	۳,۸	۳,۹
۱۶۰۰	۳,۶	۴,۰	۴,۲	۴,۴
۱۸۰۰	۳,۹	۴,۱	۴,۴	۴,۶
۲۰۰۰	۳,۵	۴,۴	۴,۷	۴,۸

نتایج نشان می‌دهد که الگوریتم پیشنهادی در کلیه حالات و بر حسب تعداد وظایف مختلف، زمان پاسخ را نسبت به روش ارائه شده قبلی کاهش می‌دهد. روش پیشنهادی به خوبی زمان اجرای کل را کاهش داده است و در این قسمت نیز دیگر معیار ارزیابی کارایی یعنی زمان پاسخ مورد مقایسه قرار گرفت و نتایج نشان می‌دهد که روش پیشنهادی این معیار را نیز به خوبی بهبود بخشیده است که بیان‌کننده در نظر گرفتن هر دو معیار به صورت توأم می‌باشد. شکل ۳ نیز این مقایسه را به خوبی نشان می‌دهد.



شکل ۳: مقایسه زمان پاسخ روش‌های مختلف

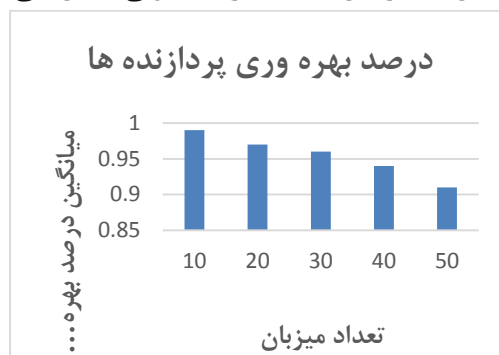
روش پیشنهادی میانگین زمان پاسخ وظایف را نیز به خوبی در مقایسه با مطالعه پیشین بهبود می‌بخشد. در این روش علاوه بر صرفه‌جویی در هزینه‌ها، تعادل بار برای سیستم‌های محاسبات توزیع شده نیز کار می‌کند و برای ایجاد تعادل در وظایف ورودی دارای کارایی مناسب است. این الگوریتم زمان اجرا را به عنوان پارامتر اصلی کیفیت سرویس در نظر می‌گیرد و منجر به بهبود معیارهای کیفی مختلف می‌گردد. نتایج به دست آمده در (شکل ۳) بیان‌گر بهره‌وری مناسب روش پیشنهادی در مقایسه با سایر روش‌ها می‌باشد. بهره‌وری بیان‌کننده نسبت زمان مشغول اجرا بودن یک منبع محاسباتی درون سیستم محاسبات ابری نسبت به کل زمان موجود را بهره‌وری یک منبع محاسباتی می‌باشد. در واقع این معیار بیان‌گر نسبت استفاده مناسب از منابع که مقدار این معیار عددی بین صفر و یک است. هر چه این عدد به یک نزدیک‌تر باشد بیان‌گر بهره‌وری بیشتر و هر چه کم‌تر از یک بوده و به صفر نزدیک باشد بیان‌گر اِتلافِ منبع محاسباتی است. با تغییر در تعداد میزبان‌ها حجم کارهای محول شده به هر میزبان تغییر می‌یابد، اما باید به این نکته توجه داشت که روش پیشنهادی در شرایطی دارای کارآمدی مناسب می‌باشد که بتواند در تمامی شرایط موجود توازن بار مناسبی بین میزبان‌های ابری برقرار کند و حجم مناسبی از وظایف را بین آن‌ها توزیع نماید. برای نشان دادن توازن مناسب از انحراف معیار متغیر بهره‌وری پردازنده‌های مختلف استفاده می‌شود. جدول ۳ کارایی روش پیشنهادی را در حالات مختلف نشان می‌دهد.

جدول ۳: بررسی میزان و انحراف معیار بهره‌وری پردازنده‌ها

تعداد میزبان‌ها	درصد بهره‌وری پردازنده‌ها	انحراف معیار
۱۰	۰.۹۹	۰
۲۰	۰.۹۷	۰.۰۲
۳۰	۰.۹۶	۰.۰۲
۴۰	۰.۹۴	۰.۰۳
۵۰	۰.۹۱	۰.۰۳

همان‌طور که نتایج ارائه شده در جدول ۳ نشان می‌دهد روش پیشنهادی در تمامی حالات مختلف تعداد میزبان‌ها، بهره‌وری پردازنده را در نسبت بالای ۹۰ درصد برقرار کرده است که این بیان‌گر کیفیت مناسب روش ارائه شده در توزیع وظایف بین پردازنده‌هاست. همان‌طور که نتایج نشان می‌دهد با افزایش اعداد میزبان‌ها با افت در بهره‌وری پردازنده‌ها مواجه هستیم که البته با توجه به افزایش منابع محاسباتی با افت ترافیک و کاهش

وظایف تخصیصی به هر پردازنده روبرو خواهیم شد که این امر منجر به کاهش اندک بهره‌وری پردازنده‌ها خواهد شد. نمودار ارائه شده در شکل ۳ نیز این مقایسه را به خوبی نشان می‌دهد.



شکل ۴: نمودار بهره‌وری پردازنده‌ها

نمودار ارائه شده در شکل ۴ نیز به خوبی بهره‌وری مناسب روش پیشنهادی را نشان می‌دهد. علاوه بر این بررسی بیشتر جدول ۳ نشان می‌دهد که درصد انحراف معیار پردازنده‌ها در میزان بهره‌وری نیز بسیار اندک می‌باشد و روش پیشنهادی به خوبی توانسته است بهره‌وری تقریباً یکسان را در بین پردازنده‌ها ایجاد کند و به خوبی شرایط پویایی مسئله را مدیریت کرده و نداشت خوبی بین وظایف و میزبان‌های ابری برقرار می‌سازد.

بحث و نتیجه‌گیری

در این پژوهش رویکردی جدید بر مبنای الگوریتم بهینه‌سازی ازدحام ذرات چندهدفه برای بهینه کردن مسئله زمان‌بندی فعالیت‌ها در محیط توزیع شده محاسباتی بر حسب اولویت دهی به اجرای وظایف بر حسب هزینه‌های اجرایی بر روی ماشین و هزینه‌های تغییر ماشین ارائه شده است. پارامترهای هزینه ماشین و هزینه تغییر ماشین به‌طور هم‌زمان در روش پیشنهادی در نظر گرفته شده اند و سپس با ایجاد توازن میان فاکتورها، بر اساس میزان اهمیت کاربر برای هر پارامتر، به زمان‌بندی وظایف در محیط محاسبات ابری پرداخته شد. نتایج آزمایشی بر این دلالت داشته که الگوریتم پیشنهادی در مقایسه با دیگر مطالعه پیشین، بهتر عمل می‌کند. نتایج نشان می‌دهد که الگوریتم پیشنهادی هزینه‌های اضافی را به خوبی کاهش می‌دهد. این روش تعادل بار برای سیستم‌های محاسبات توزیع ناهمگن کار می‌کند و برای ایجاد تعادل در وظایف دارای کارایی مناسبی می‌باشد.

مراجع

[۱] کامل طباطبائی فریضی، سید رضا، هاشمی، سیده طیبه، ۱۳۹۵، الگوریتم تعادل بار وظایف وابسته در محاسبات ابری با الهام از رفتار زنبور عسل - کنفرانس بین‌المللی پژوهش‌های نوین در علوم مهندسی.

[۲] اسمعیلی، محمد، میرزایی، عباس، ۱۳۹۴، زمان‌بندی وظایف در رایانش ابری با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات بهبود یافته، کنفرانس بین‌المللی یافته‌های نوین پژوهشی در مهندسی برق و علوم کامپیوتر.

[3] S. Sowndarya, L. Ben, Communication Augmented Latest Possible Scheduling for cloud computing with delay constraint and task dependency, 2016

- [4] Huang, Daochao, et al. "Energy-aware virtual machine placement in data centers." Global Communications Conference (GLOBECOM), 2012 IEEE. IEEE, 2012.
- [5] A. Oprescu and T. Kielmann, "Bag-of-tasks scheduling under budget constraints," in Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on, 2010, pp. 351-359.
- [6] R. Van den Bossche, K. Vanmechelen, and J. Broeckhove, "Online costefficient scheduling of deadline-constrained workloads on hybrid clouds," *Future Generation Computer Systems*, vol. 29, pp. 973-985, 2013.
- [7] J. Li, J. Peng, Z. Lei, W. Zhang, "An Energy-Efficient Approach Based on Private Clouds", *Journal of Information & Computational Science*, april 2011.
- [8] S. K. Garg, C. S. Yeo, A. Anandasivam, R. Buyya, "Energy-Efficient Scheduling of HPC Applications in Mobile Cloud Computing Environments", Elsevier publications, September 2013.
- [9] R. Chang, J. Chang, Po-Sheng Lin, "An ant algorithm for balanced job scheduling in grids", *Future Generation Computer Systems*, june 2008.
- [10] U. gurean, "Job Scheduling Algorithm based on Dynamic Management of Resources Provided by Mobile Computing System.", feb 2010.
- [11] G. Gharoonifar, Fahime Moeidarbari, Hossein Deldari, Anahita Morvaridi, "Scheduling of scientific workflows using a chaos-genetic algorithm", Elsevier publications, 2014.
- [12] U. Deshpande, Y. You, D. Chan, N. Bila, K. Gopalan, (2014) "Fast server deprovisioning through scatter-gather live migration of virtual machine", *IEEE Cloud*.
- [13] M. Ramachandran, V. Chang, (2016) "Towards performance evaluation of cloud service providers for cloud data security", *International Journal of Information Management*, Vol. 36, no. 4, pp, 618-625.
- [14] M. Galloway, L. Gabriel, S. Vrbsky, (2015) "Performance Metrics of Virtual Machine Live Migration", *Mobile Cloud Computing (CLOUD) 2015 IEEE 8th International Conference on*. IEEE, 2015.
- [15] Zhang F. Chen J. Chen H. Zang B, (2011) "CloudVisor: retrofitting protection of virtual machines in multi-tenant cloud with nested virtualization", *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, 2011.
- [16] Lin C. C. Jian Z. D. Hsu C. H. (2014) "A Strategy of Service Quality Optimization for Live Virtual Machine Migration", *Proc. the IEEE 17th Int. Conf. Computational Science and Engineering*, vol. 566, pp. 1308-1313.
- [17] Paolo B. Rebecca M. Sajal K.' Mobile social networking middleware: A survey'. *Mobile social networking middleware: A survey, Pervasive and Mobile Computing*, pp-1_33. 2015.
- [18] M. Galloway, L. Gabriel, S. Vrbsky, (2015) "Performance Metrics of Virtual Machine Live Migration", *Mobile Cloud Computing (CLOUD) 2015 IEEE 8th International Conference on*. IEEE, 2015.
- [19] Zhang F. Chen J. Chen H. Zang B, (2011) "CloudVisor: retrofitting protection of virtual machines in multi-tenant cloud with nested virtualization", *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, 2011.

- [20] Lin C. C. Jian Z. D. Hsu C. H. (2014) "A Strategy of Service Quality Optimization for Live Virtual Machine Migration", Proc. the IEEE 17th Int. Conf. Computational Science and Engineering, vol. 566, pp. 1308-1313.
- [21] Y. Liu, H. Shao, W. Jing, Zhaowen Qiu, Multi-DAGs Scheduling Integrating with Security and Availability in Mobile Cloud Environment, Chinese Journal of Electronics, Vol. 24, no. 4, 2015.
- [22] Zhang J. Ren F. Shu R. (2016) "Guaranteeing Delay of Live Virtual Machine Migration by Determining and Provisioning Appropriate Bandwidth", IEEE Trans. Comput. vol. 65, no. 9, pp. 2910-2917.
- [23] Shafi'i M. A., Latiff M. S., Gaddafi A., Madni S. H., Secure Scientific Applications Scheduling Technique for Cloud Computing Environment Using Global League Championship Algorithm, PLoS One Journal. Vol. 6, no. 11, 2016.
- [24] P. Kumar, A. Verma, Scheduling using improved genetic algorithm in cloud computing for independent tasks, ICACCI '12 Proceedings of the International Conference on Advances in Computing, Communications and Informatics, Pages 137-142, 2012.
- [25] M. A. Tawfeek, A. El-Sisi, A. E. Keshk, F. A. Torkey, Cloud task scheduling based on ant colony optimization, Computer Engineering & Systems (ICCES), 2013 8th International Conference on, 2013.