

موضوع: بررسی الگوریتم‌های مبتنی بر توازن بار و تخصیص منابع

۱-امین کریم الله دهکردی

(کارشناسی ارشد مهندسی کامپیوتر)

۲-دکتر گلناز آقایی

(دکتری تخصصی سیستم‌های نرم افزاری و استادیار دانشگاه)

۱-۱-چکیده

در سال‌های اخیر، با گسترش استفاده از فناوری محاسبات ابری و افزایش تعداد کاربران در سیستم‌های ابری سیار، مدیریت موثر منابع و توازن بار به یک چالش اساسی تبدیل شده است. این چالش‌ها به‌خصوص در محیط‌های ابری سیار که دارای محدودیت‌هایی مانند انرژی محدود و نوسانات بالا در تقاضای منابع هستند، بیشتر نمود پیدا می‌کنند. بسیاری از روش‌های مرجع در این زمینه، از جمله الگوریتم‌های مبتنی بر توازن بار و تخصیص منابع، معمولاً با مشکلاتی نظیر مصرف زیاد انرژی، زمان پاسخ طولانی، و عدم توازن مناسب بار مواجه هستند. این موضوعات منجر به کاهش کارایی کلی سیستم‌های ابری و افزایش هزینه‌ها می‌شود. در این مقاله، ما با مروری بر آخرین رویکردها، الگوریتم‌های زمانبندی و تخصیص منابع را مورد بحث و مقایسه قرار داده و طبقه‌بندی از این الگوریتم‌ها ارائه می‌کنیم.

کلیدواژگان: زمانبندی، تخصیص منابع، توازن بار

۱-۲-مقدمه

رایانش ابری مدلی بر پایه ی شبکه‌های بزرگ کامپیوتری مانند اینترنت است که الگویی تازه برای عرضه، مصرف و تحویل سرویس‌های فناوری اطلاعات و سایر منابع اشتراکی رایانشی با بکارگیری اینترنت را ارائه می‌کند. رایانش ابری راهکارهایی برای ارائه خدمات فناوری اطلاعات به شیوه‌های مشابه با صنایع همگانی پیشنهاد می‌کند. دلیل استفاده از واژه ی ابر این است که پردازش ابری جزئیات فنی اش را از دید کاربران پنهان می‌سازد و لایه ای از انتزاع را بین این جزئیات فنی و کاربران به وجود می‌آورد. به عنوان مثال آنچه یک ارائه دهنده ی سرویس نرم افزاری در رایانش ابری ارائه می‌کند، برنامه‌های کاربردی تجاری برخط است که از طریق مرورگر وب یا نرم افزارهای دیگر به کاربران ارائه می‌شود. نرم افزارهای کاربردی و اطلاعات روی سرورها ذخیره می‌گردند و براساس تقاضا در اختیار کاربران قرار می‌گیرد. جزئیات از دید کاربر مخفی می‌مانند و کاربران نیازی به تخصص یا کنترل در مورد فناوری زیر ساخت ابری که از آن استفاده می‌کنند ندارند. رایانش ابری را گروهی تغییر الگوواره ای می‌دانند که دنباله روی تغییری است که در اوایل دهه ۱۹۸۰ از مدل رایانه بزرگ به مدل کارخواه-کارساز صورت گرفت. امروزه سازمان‌ها و شرکت‌ها سرویس‌هایی را در اختیار کاربران خود قرار می‌دهند که از کار افتادن سرورها و به طبع سرویس‌های آنها، زبان‌های مالی را به سازمان وارد می‌کند. به طور مثال سازمان از طریق وب سایت خود خدماتی را در اختیار کاربران قرار می‌دهد، مانند بانک که از طریق وب سایت امکان بررسی حساب و نقل و انتقال مالی را به مشتریان

ارائه می‌کند. در صورت از کار افتادن یا افت سرویس دهی، باعث زیان‌های مالی و نارضایتی مشتریان خواهد شد. هدف از انجام پژوهش پیش رو بررسی الگوریتم‌های مبتنی بر توازن بار و تخصیص منابع می‌باشد.

۳-۱- بیان مسئله

در محاسبات ابری سیار، دستگاه‌های سیار به دلیل محدودیت‌هایی از قبیل توان پردازشی پایین، عمر محدود باتری، و پهنای باند محدود، نیازمند تخصیص هوشمندانه منابع ابری برای اجرای وظایف محاسباتی هستند. این امر به ویژه در سناریوهایی که تعداد زیادی از دستگاه‌ها به طور همزمان به منابع ابری نیاز دارند، اهمیت بیشتری پیدا می‌کند. از طرفی، سرویس‌دهنده‌های ابری نیازمند آن هستند که منابع خود را به گونه‌ای مدیریت کنند که هم کارایی سیستم بالا بماند و هم مصرف انرژی به حداقل برسد. با توجه به این محدودیت‌ها، مسئله توازن بار و تخصیص منابع به یک مسئله چالش‌برانگیز و پیچیده تبدیل شده است که نیازمند الگوریتم‌ها و روش‌های بهینه‌سازی موثر است.

روش‌های مرجع در مدیریت منابع و توازن بار، معمولاً با رویکردهای مختلفی سعی در حل این مسئله داشته‌اند. برخی از این روش‌ها بر پایه الگوریتم‌های کلاسیک توازن بار مانند Round Robin و Least Connections عمل می‌کنند که علی‌رغم سادگی، قادر به پاسخگویی به نیازهای پیچیده محیط‌های ابری سیار نیستند. این روش‌ها به طور عمده با مشکلاتی نظیر افزایش مصرف انرژی، عدم تطابق با نوسانات لحظه‌ای تقاضا، و ناتوانی در دستیابی به توازن بار بهینه روبه‌رو هستند. در برخی دیگر از رویکردها، الگوریتم‌های بهینه‌سازی مبتنی بر جستجوی فراابتکاری مانند الگوریتم ژنتیک (GA) و الگوریتم بهینه‌سازی ازدحام ذرات (PSO) پیشنهاد شده‌اند که با هدف بهبود عملکرد و کاهش هزینه‌ها طراحی شده‌اند. هرچند این الگوریتم‌ها توانسته‌اند تا حدودی بهبودهایی را در زمینه مدیریت منابع ارائه دهند، اما همچنان با چالش‌هایی نظیر زمان محاسباتی بالا و پیچیدگی پیاده‌سازی مواجه هستند.

مسئله اصلی این پژوهش از اینجا نشأت می‌گیرد که با وجود پیشرفت‌های انجام شده در زمینه مدیریت منابع و توازن بار در محاسبات ابری سیار، هنوز هم نقاط ضعف قابل توجهی وجود دارد که منجر به کاهش کارایی کلی سیستم می‌شود. این نقاط ضعف شامل مصرف انرژی بالا، زمان پاسخ طولانی، و عدم توازن مناسب بار در ماشین‌های مجازی است. با توجه به محدودیت‌های دستگاه‌های سیار و نیاز به بهره‌وری بیشتر از منابع موجود، توسعه رویکردهای نوآورانه‌ای که بتوانند به طور موثر این چالش‌ها را مدیریت کنند، ضروری به نظر می‌رسد.

۴-۱- پیشینه پژوهش

یانگ-کوو لین و همکارانش در سال ۲۰۱۵ یک الگوریتم GA بهینه جهت تخصیص منابع محاسباتی در یک سیستم تولید ابرایه کردند که هدف کاهش هزینه و زمان پاسخ به درخواست مشتری و افزایش کارایی و عملکرد بهتر است ولی در این تحقیق عملکرد makespan (زمان تکمیل برنامه گردش کار: زمان شروع اجرای اولین وظیفه تا اتمام اجرای آخرین وظیفه) و بهره‌وری در نظر گرفته نشده بود.

محققین این مقاله در سال ۲۰۱۵ یک مدل برای استفاده از انرژی منابع آگاه بر اساس ABC به صورت پویا مطرح کردند که به نحو احسن مدیریت منابع ابرو افزایش استفاده از این منابع را انجام می‌دهد و باعث صرفه جویی در انرژی، کاهش زمان اجرای برنامه‌های ارائه شده به ابرو و عملکرد بهتر می‌شود ولی حجم کار در گره در نظر گرفته نشده بود. این الگوریتم با سه الگوریتم ERU و ACO و FFD مقایسه شده است.

فهیمة رضانی و همکارانش در سال ۲۰۱۴ یک روش متعادل کردن بار سیستم بر اساس task و انتقال VM با استفاده از بهینه‌سازی ازدحام ذرات در ابررایانه به نام TBSLB-PSO ارائه کردند که کارشان باعث کاهش زمان، متوقف نشدن فرآیند مهاجرت ماشین‌های مجازی در طول اجرای الگوریتم، کاهش زمان انتقال وظایف، کاهش هزینه، کاهش مصرفی در زمان اجر شدن و عیب کار این بود که حافظه زیادی از دست می‌رفت.

محققین این مقاله در سال ۲۰۱۴ یک معماری جدید از الگوریتم ژنتیک برای laas و پلت فرم ابر محاسبات که در آن تخصیص ماشین مجازی انجام می‌گیرد پیشنهاد داده اند و سه الگوریتم GA-BPNN و GA-ELNN و GA-JNN باهم مقایسه شدند و GA-JNN بهترین زمان و بهبود بهره‌وری رانشان داده است ولی روش پیشنهادی به پیش بینی میزان بار میزبان در گذشته بستگی دارد.

۵-۱- چالش‌های موجود در زمینه تخصیص منابع در محاسبات ابری

۱. بهینه‌سازی مصرف انرژی:

یکی از چالش‌های اصلی در تخصیص منابع محاسباتی، کاهش مصرف انرژی است. با توجه به رشد روزافزون تعداد دستگاه‌های سیار و درخواست‌های محاسباتی، نیاز به روش‌هایی که بتوانند مصرف انرژی را بهینه کنند، احساس می‌شود. در این زمینه، الگوریتم‌های کلونی مورچه و بهینه‌سازی ازدحام ذرات توانسته‌اند بهبودهای قابل توجهی ارائه دهند.

۲. توازن بار:

توازن بار یکی دیگر از چالش‌های حیاتی است که در تخصیص منابع محاسباتی باید به آن توجه شود. بسیاری از روش‌های سنتی نمی‌توانند توازن بار را به صورت موثر برقرار کنند، که این امر منجر به افزایش زمان پاسخگویی و کاهش کارایی سیستم می‌شود. الگوریتم‌های بهینه‌سازی کلونی مورچه و ژنتیک می‌توانند در این زمینه بهبودهای محسوسی ایجاد کنند.

۳. کاهش زمان پاسخ:

کاهش زمان پاسخگویی به درخواست‌ها از اهمیت بالایی برخوردار است، خصوصاً در محیط‌های ابری که نیاز به پاسخگویی سریع به کاربران دارند. استفاده از الگوریتم‌های بهینه‌سازی نظیر بهینه‌سازی ازدحام ذرات می‌تواند در کاهش زمان پاسخ موثر باشد.

۴. بهینه‌سازی مشترک منابع:

یکی از چالش‌های پیچیده‌تر، تخصیص بهینه منابع به صورت مشترک بین محاسبات و شبکه است. این موضوع به دلیل وابستگی شدید بین مصرف منابع و پهنای باند شبکه نیازمند روش‌های پیچیده‌تری است. الگوریتم‌های ژنتیک و بهینه‌سازی کلونی مورچه‌ها می‌توانند در این زمینه مؤثر واقع شوند.

۵. مدیریت بهینه انرژی و بهره‌وری:

بسیاری از روش‌های موجود هنوز نمی‌توانند به صورت همزمان بهره‌وری انرژی و عملکرد سیستم را بهینه کنند. این چالش نیازمند راهکارهایی است که بتوانند میان بهره‌وری انرژی و کارایی سیستم توازن برقرار کنند.

در مجموع، این چالش‌ها نشان می‌دهند که هنوز فضای زیادی برای تحقیق و توسعه در زمینه تخصیص منابع در محیط‌های محاسبات ابری وجود دارد و الگوریتم‌های پیشرفته می‌توانند در این راستا نقش مهمی ایفا کنند.

۶-۱- انواع الگوریتم‌های توازن بار

بر اساس اینکه چه کسی روند توازن بار را آغاز می‌کند، الگوریتم‌های توازن بار می‌توانند در سه دسته بندی قرار گیرند:

فرستنده آغاز کننده باشد: اگر الگوریتم‌های توازن بار توسط فرستنده آغاز شده باشد.

گیرنده آغاز کننده باشد: اگر الگوریتم‌های توازن بار توسط گیرنده آغاز شده باشد.

مقارن: ترکیبی از هر دو حالت بالا

بر اساس حالت جاری سیستم، الگوریتم‌های توازن بار می‌توانند در دو دسته قرار بگیرند:

ایستا: بستگی به وضعیت جاری سیستم ندارد و نیاز به دانش و آگاهی قبلی، از سیستم می‌باشد.

پویا: نسبت به وضعیت جاری سیستم روی توازن بار تصمیم گیری می‌شود.

روش‌های پیاده سازی الگوریتم‌های توازن یا تعادل ابر می‌تواند به صورت متمرکز، توزیعی و سلسله مراتبی باشد. در روش متمرکز تمامی وظایف زمان بندی و تعادل بار بر روی یک گره است که این خاصیت باعث کاهش زمان آنالیز کردن گره‌ها می‌شود ولی سربار بسیار زیادی روی گره ی توازن دارد. در شیوه ی توزیعی هر گره یک پایگاه دانش محلی از وضعیت بار دارد و این ویژگی موجب تحمل پذیری در برابر خرابی می‌شود. و تکنیک سلسله مراتبی با ساختار درخت مدل سازی می‌شود که هر گره ی پدر مسئول برقرار کردن تعادل بین گره‌های فرزند خود است .

توازن بار ایستا، یک کار را به یک پردازنده یا گره ثابت تخصیص می‌دهد. هر زمانی که سیستم دوباره راه اندازی شود، از همان پردازنده ی وظیفه ی متصل شده، بدون در نظر گرفتن تغییراتی که ممکن است در طول دوران حیات سیستم رخ دهد، استفاده می‌شود. ممکن است تخصیص وظیفه به همان پردازنده، نتیجه ندهد، اما اختصاص کارهایی که تازه از راه می‌رسند، در یک ترتیب یا مدل ثابت است. کارایی ماشین‌های مجازی، در زمان رسیدن کار تعیین می‌گردد .

پردازنده ی رئیس حجم کاری را به دیگر پردازنده‌های برده بر اساس کارایی آنها، تخصیص می‌دهد. کار اختصاص داده شده به این ترتیب توسط پردازنده برده اجرا شده و نتیجه به پردازنده رئیس برگردانده می‌شود.

وظایف در الگوریتم‌های توازن بار ایستا بر اساس توانایی گره برای پردازش درخواست جدید تخصیص می‌یابند. این فرآیند صرفاً بر اساس دانش قبلی از خواص گره است که می‌تواند شامل: قدرت پردازش گره، حافظه، ظرفیت ذخیره سازی و جدیدترین کارایی ارتباطاتی شناخته شده باشد. الگوریتم‌های ایستا معمولاً تغییرات پویای زمان اجرای این صفات را در نظر نمی‌گیرند. علاوه بر این، این نوع الگوریتم‌ها نمی‌توانند با تغییرات بار در زمان اجرا سازگار شوند هدف آنها به حداقل رساندن زمان اجرای کار و محدود کردن سربار ارتباطات و تأخیر است.

در الگوریتم‌های توازن بار پویا حجم کاری، در زمان اجرا میان پردازنده‌ها، توزیع می‌شود. پردازنده رئیس، فرآیند جدید را بر اساس اطلاعات جمع آوری شده ی جدید به پردازنده ی برده اختصاص می‌دهد.

بیشتر این الگوریتم‌ها بر روی ترکیبی از اطلاعاتی که از قبل درباره ی گره‌ها در ابر جمع آوری شده و خصوصیات زمان اجرای جمع آوری شده، تکیه می‌کنند، به طوری که گره‌های انتخابی اجزای وظایف را پردازش می‌کنند.

این الگوریتم‌ها به نظارت مستمر گره‌ها و جریان وظایف نیاز دارند و معمولاً برای پیاده‌سازی سخت‌تر می‌باشند. اما این الگوریتم‌ها دقت بالاتری دارند و می‌توانند نتایج توازن بار بهینه‌تری تولید کنند. این الگوریتم‌ها اجازه‌ی پیش‌دستی پردازش‌ها را می‌دهد که در روش توازن بار ایستا، پشتیبانی نمی‌شود.

یک مزیت مهم توازن بار پویا این است که تصمیم توازن بار بر اساس حالت فعلی سیستم است که به بهبود کارایی کلی سیستم با مهاجرت پویای بار، کمک می‌کند. یک استراتژی پویا معمولاً چندین بار اجرا می‌شود و ممکن است یک کار زمانبندی شده را دوباره به یک گره جدید بر اساس حالت پویای محیط سیستم، اختصاص دهد.

توازن بار پویا می‌تواند به دو روش متفاوت انجام شود: توزیع شده و توزیع نشده. در نوع توزیع شده، الگوریتم‌های توازن بار پویا توسط تمام گره‌های موجود در سیستم اجرا می‌شود و وظیفه توازن بار میان تمامی آن‌ها تقسیم می‌شود. تعامل میان گره‌ها برای رسیدن به توازن بار می‌تواند به دو شکل باشد: همکاری و بدون همکاری.

در فرم اولی گره‌ها در کنار هم برای رسیدن به یک هدف مشترک کار می‌کنند. برای مثال، بهبود زمان پاسخ کل، و در فرم دومی هر گره به صورت مستقل در جهت یک هدف محلی کار می‌کند مثل بهبود زمان پاسخ یک کار محلی. الگوریتم‌های توازن بار پویا با ماهیت توزیع شده، پیام‌های بیشتری را نسبت به نوع توزیع نشده تولید می‌کنند به‌طوری‌که هر گره درون سیستم نیاز به برقراری ارتباط با هر گره دیگر دارد. یک مزیت این روش در این است که اگر یک یا تعدادی از گره‌ها درون این سیستم خراب شوند، باعث نمی‌شود که کل فرآیند توازن بار متوقف شود، ولی تا حدی بر روی کارایی سیستم تأثیر می‌گذارد [۹]. توازن بار پویای توزیع شده می‌تواند روی سیستمی که در آن هر گره به تبادل اطلاعات با گره‌های دیگر درون سیستم نیاز دارد، فشار زیادی را ایجاد کند. این روش زمانی نسبت به روشی که بیشتر گره‌ها به صورت انفرادی با تعامل اندکی با گره‌های دیگر کار می‌کنند، بهتر است. در روش توزیع نشده، یک گره و یا یک گروهی از گره‌ها وظیفه توازن بار را انجام می‌دهند.

الگوریتم‌های توازن بار پویای توزیع نشده می‌تواند به دو نوع متمرکز و نیمه توزیع شده باشد. در نوع متمرکز، الگوریتم توازن بار فقط بر روی یک گره (سیستم) گره مرکزی) اجرا می‌شود. این گره به تنهایی وظیفه توازن بار کل سیستم را به عهده دارد. گره‌های دیگر فقط با گره مرکزی تعامل دارند. در نوع نیمه توزیع شده، گره‌های سیستم به خوشه‌هایی تقسیم شده‌اند که در آن توازن بار هر خوشه به صورت متمرکز می‌باشد. به وسیله‌ی تکنیک‌های مناسب برای هر خوشه یک گره مرکزی انتخاب می‌شود تا توازن بار را در خوشه مراقبت کند. از این رو توازن بار کل سیستم توسط گره‌های مرکزی هر خوشه انجام می‌شود توازن بار پویای متمرکز، پیام‌های کمتری را دریافت می‌کند. از این رو از تعداد کل تعاملات درون سیستم نسبت به نوع نیمه توزیع شده، کاسته می‌شود. با این حال الگوریتم‌های متمرکز می‌توانند روی گره مرکزی گلوگاه ایجاد کنند و هم چنین روند توازن بار هنگامی که گره مرکزی سقوط می‌کند، بی‌نتیجه می‌شود. بنابراین این الگوریتم برای شبکه‌هایی با اندازه‌ی کوچک مناسب‌تر است.

۷-۱- الگوریتم‌های تکاملی

هوش جمعی شاخه‌ای از پژوهش بر اساس جمعیت است که در آن مدل‌های جمعیتی در برابر عوامل تداخل یا ازدحام می‌توانند خود را سازمان‌دهی کنند. کلونی مورچه، ازدحام پرندگان و یا زنبورها نمونه‌هایی ساده از سیستم‌های جمعیتی است. دیگر نمونه‌ای از هوش جمعی کلونی زنبور عسل در اطراف کندو است. هوش کلونی زنبور عسل (ABC) الگوریتمی بهینه‌سازی شده بر اساس رفتار هوشمندانه جمعیت زنبور عسل است.

تلاش‌های زیادی برای مدل کردن رفتارهای خاص و هوشمندانه تجمع زنبورهای عسل انجام گرفته است Tereshko و Loengarov کلونی زنبور را به عنوان یک سیستم پویا در نظر گرفتند که از محیط اطراف اطلاعات جمع‌آوری کرده و رفتار خود را بر اساس این اطلاعات به دست آمده تنظیم می‌نماید. آن‌ها یک ایده رباتی با توجه به رفتار کاوشی زنبورها مطرح کردند. غالباً این ربات‌ها به صورت فیزیکی و عملکردی یکسان هستند. در نتیجه هر ربات می‌تواند به طور تصادفی جایگزین دیگر ربات‌ها گردد. تجمع، قابلیت تحمل خطا را دارد. با رخ دادن خطا در یک عامل کار کل سیستم مختل نخواهد شد. ربات‌های مجزا، مانند حشرات، دارای قابلیت‌ها و توانایی‌های محدودی هستند. همچنین دانش محدودی از محیط دارند. به عبارتی دیگر تجمع (ازدحام)، هوش جمعی هم کارانه را بهبود می‌دهد. همچنین این آزمایش نشان می‌دهد که ربات‌های حشره مانند در انجام وظایف حقیقی ربات‌ها، موفق هستند. به علاوه آن‌ها یک مدل کمینه از رفتار کاوشگرانه زنبورها ارائه دادند. این مدل شامل سه مؤلفه مهم می‌باشد:

(1) منبع غذایی

(2) زنبورهای کارگر

(3) زنبورهای غیر کارگر.

این مدل دو نوع رفتار را در برمی‌گیرد: سربازگیری برای یک منبع شهد و ترک منبع. Teodorovic الگوریتم کلونی زنبور عسل مانند سایر الگوریتم‌های هوش ازدحامی مرتبط بر رفتار تصادفی المان‌های آن است و برای حل مسائل بهینه‌سازی کاربرد دارد. بسیاری از الگوریتم‌های هوش ازدحامی با الهام گرفتن از طبیعت ایجاد شده‌اند. برخی از کاربردهای الگوریتم بهینه‌سازی زنبور عسل در علوم مهندسی به صورت زیر است:

- آموزش شبکه عصبی برای الگو شناسی
- زمان بندی کارها برای ماشین‌های تولیدی
- دسته بندی اطلاعات
- بهینه سازی طراحی اجزای مکانیکی
- بهینه سازی چندگانه
- میزان کردن کنترل کننده‌های منطق فازی برای ربات‌های ورزشکار

بسیاری از مسائل به روش‌های معمول ریاضی قابل حل نیستند و یا حل کردن آن‌ها زمان بسیار زیادی را می‌طلبد. در این نوع از مسائل ما به دنبال پیدا کردن یک نقطه بهینه در مسئله هستیم که اصطلاحاً به آن نقطه، نقطه بهینه می‌گوییم. نقطه بهینه زمانی به دست می‌آید که ما کمترین خطا در مسئله را داشته باشیم. الگوریتم‌هایی تصادفی مانند الگوریتم ژنتیک و الگوریتم‌های تکاملی برای حل مسائل بهینه‌سازی استفاده می‌شوند. یکی دیگر از روش‌های حل مسائل بهینه‌سازی الگوریتم‌های هوش ازدحامی است که الگوریتم زنبور عسل از جمله این دست الگوریتم‌ها است. الگوریتم زنبور (Bee Algorithm) یک الگوریتم گروهی مبتنی بر جستجو است. این الگوریتم شبیه سازی رفتار جستجوی غذای گروه‌های زنبور عسل است. در نسخه ابتدایی این الگوریتم، الگوریتم نوعی از جستجوی محلی انجام می‌دهد که با جستجوی تصادفی ترکیب شده و می‌تواند برای بهینه‌سازی ترکیبی یا بهینه‌سازی تابعی استفاده شود. این الگوریتم نیز مانند سایر الگوریتم‌های هوش ازدحامی از دو روش اکتشاف و استخراج استفاده می‌کند. زنبورهای کارگر وظیفه استخراج و زنبورهای

ناظر وظیفه اکتشاف را به عهده دارند. زنبورهای کارگر در اطراف یک منطقه (گل‌های پیداشده یا منطقه‌ای که شامل جواب مسئله است) به دنبال جواب بهینه می‌گردند و زنبورهای ناظر با رفتار تصادفی به دنبال پیدا کردن مناطق جدید هستند (گل‌های جدید).

۸-۱- الگوریتم‌های توازن بار دو مرحله ای: OLB + LBMM

یک الگوریتم زمانبندی دو مرحله ای پیشنهاد شده است که الگوریتم‌های زمانبندی (OLB فرصت توازن بار (LBMM) توازن بار (Min-Min) را برای استفاده از اجرای بهتر و حفظ توازن بار سیستم، ترکیب می‌کند. الگوریتم زمانبندی OLB هر گره را برای رسیدن به هدف توازن بار، در حالت کار نگه

می‌دارد و الگوریتم زمانبندی LBMM برای کاهش زمان اجرای هر وظیفه روی گره استفاده می‌شود، در نتیجه زمان اجرای کل را کاهش می‌دهد. محیط استفاده از این الگوریتم شبکه‌های رایانش ابری سه سطحی است و در آن، معیار کارایی و بهره برداری از منابع در نظر گرفته شده است. این الگوریتم ترکیبی، در استفاده ی موثر از منابع کمک می‌کند و بهره وری را افزایش می‌دهد نتایج بهتری را نسبت به کاوش زنبور عسل، نمونه برداری تصادفی و خوشه بندی فعال ارائه می‌دهد [۱۱].

در الگوریتم LC به صورت پویا تعداد اتصالات همه ی سرویس دهندگان را بررسی کرده و از میان آنها سرویس دهنده ای که کمترین اتصالات را داشته باشد، برای مقصد وظیفه ی بعدی انتخاب می‌شود.

الگوریتم WLC یک الگوریتم خوب زمان بندی از نوع پویاست که در واقع همان الگوریتم LC است که نسبت به توان پردازشی، وزنی برای آن در نظر گرفته می‌شود. حاصل ضرب این وزن در تعداد اتصالات هر سرویس دهنده بیان کننده ی مناسب یا نامناسب بودن آن سرویس دهنده برای وظیفه ی بعدی است. از جمله مشکلات این الگوریتم می‌توان به این موارد اشاره کرد :

۲) ادمین به طور ساختگی وزن هر سرور را با توجه به عملکرد گره ی سرور و شرایط بار تخمین می‌زند .

۲) وزن به طور دستی توسط ادمین پیکر بندی شده است.

۲) این الگوریتم اطلاعات حداقل اتصال نظیر فرکانس پردازنده، تعداد CPU، مصرف کننده ی پردازنده، اندازه حافظه، و جریان شبکه را در بر نمی‌گیرد.

۹-۱- روش کولونی مورچه‌ها

زیرکی گروه مورچه‌های مصنوعی به ویژه بهینه سازی منصوب به محاسبات و (ACO) کولونی مورچه‌ها نمونه رفتار جدید برای حل مسائل بهینه سازی ترکیبی می باشد و بر اساس اصولی است که می تواند رفتار سیستم‌های طبیعی را کنترل کند. هنگامی از موجودات هوشمند برای مسائل شبکه بهره گرفته (Agents) می شود که یک گروه از برنامه‌ها با هم کار می کنند و می توانند برای کاربردهایی مانند یافتن کوتاهترین مسیر، مسیریابی، توازن بار و مدیریت و مانند آن استفاده شوند.

ACO یک سیستم چند عامله است که در آن ارتباط بین مورچه‌های مصنوعی در رفتار ی با باز خورد مثبت که مورچه‌ها را به سمت یک راه حل مطلوب راهنمایی می‌کند، نتیجه می‌شود. اطلاعات فرمون (شبه سازی شده ی مواد شیمیایی که مورچه‌ها در مسیر واقعی ذخیره کردند). به یالهای گراف اختصاص می‌یابد. در مقایسه با الگوریتم ژنتیک، ACO مزایایی همچون اجازه ی بازخورد مثبت، محاسبات توزیع شده و جستجوی اکتشافی حریصانه را دارد.

۱۰-۱- نتیجه گیری

هر یک از این الگوریتم‌ها نقاط قوت و ضعفی دارند که نشان‌دهنده‌ی چالش‌های اساسی در این حوزه است. این چالش‌ها نشان می‌دهند که هنوز فضای زیادی برای بهبود وجود دارد. در ادامه، چالش‌های کلیدی قابل حل مورد بحث قرار گرفته‌اند:

۱. مقیاس‌پذیری الگوریتم‌ها:

○ بسیاری از الگوریتم‌های ارائه‌شده مانند الگوریتم‌های Mean-Min, Sufferage, Min-min، ژنتیک و خفاش، در مقیاس‌های بزرگ قابلیت استفاده ندارند. این محدودیت نشان‌دهنده نیاز به توسعه الگوریتم‌هایی است که بتوانند با افزایش تعداد وظایف و منابع، به صورت کارآمد عمل کنند. بهبود مقیاس‌پذیری از اهمیت ویژه‌ای برخوردار است، زیرا در محیط‌های رایانش ابری، تعداد زیادی از وظایف باید به طور همزمان پردازش شوند.

۲. مصرف انرژی:

○ برخی از الگوریتم‌ها مانند "Priority Based Performance Improved Algorithm" مصرف انرژی بالایی دارند. کاهش مصرف انرژی یکی از اهداف اصلی در رایانش ابری است، زیرا مصرف کمتر انرژی نه تنها هزینه‌ها را کاهش می‌دهد بلکه به حفاظت از محیط زیست نیز کمک می‌کند. توسعه الگوریتم‌هایی که بهینه‌سازی انرژی را در کنار زمان‌بندی وظایف به صورت همزمان مد نظر قرار دهند، یکی از چالش‌های مهم قابل حل است.

۳. پیچیدگی الگوریتم‌ها:

○ الگوریتم‌های پیچیده مانند MOP و الگوریتم مورچگان با مشکلاتی از جمله واگرایی و پیچیدگی مواجه هستند. این پیچیدگی می‌تواند منجر به عدم کارایی در محیط‌های واقعی شود. به همین دلیل، یکی از چالش‌های کلیدی، کاهش پیچیدگی الگوریتم‌ها بدون از دست دادن کارایی و بهینه‌سازی است.

۴. تعامل با محیط‌های پویا:

○ برخی از الگوریتم‌ها، مانند الگوریتم‌های مبتنی بر هزینه، نیازمند بهبود برای تعامل با محیط‌های پویا مانند تغییرات دینامیک در محیط ابری و ویژگی‌های QoS هستند. این مسئله بهبودهایی را در زمینه کنترل و مدیریت وظایف پویا در رایانش ابری می‌طلبد.

۵. استفاده مؤثر از منابع:

○ بسیاری از الگوریتم‌ها بر روی استفاده مؤثر از منابع تأکید دارند اما در عمل ممکن است به دلیل نادیده گرفتن پارامترهای خاصی، بهینه‌سازی کامل را نتوانند فراهم کنند. برای مثال، الگوریتم‌های مورچگان و نهنگ در بعضی از شرایط خاص، خوشه‌های ناهمگن را در نظر نمی‌گیرند. بهبود استفاده از منابع در محیط‌های ناهمگن و پویا یکی از چالش‌های قابل حل است.

۶. عدم وجود ارزیابی واقعی:

○ برخی الگوریتم‌ها، مانند MOP، در محیط‌های واقعی ارزیابی نشده‌اند. عدم ارزیابی در محیط واقعی می‌تواند منجر به ناکارآمدی الگوریتم‌ها در شرایط عملیاتی شود. بنابراین، یکی از چالش‌های مهم، اجرای الگوریتم‌ها در محیط‌های واقعی و بررسی عملکرد آنها در شرایط مختلف است.

۷. تداخل الگوریتم‌ها:

o ترکیب الگوریتم‌ها مانند ترکیب Min-min و Sufferage گاهی منجر به تداخل و کاهش کارایی در مقیاس‌های بزرگ می‌شود. مدیریت تداخل و بهینه‌سازی ترکیب الگوریتم‌ها برای بهبود کارایی و کاهش زمان پاسخ یکی از چالش‌های مهمی است که باید حل شود.

در روش پایه الگوریتم کلونی مورچه برای انتخاب کارها از یک رابطه ایستا استفاده می‌شود به‌طوری که در ابتدا برای تمام کارها مقدار فرمون محاسبه می‌شود و بر اساس آن تا انتها کارها انتخاب می‌شوند ولی در روش پیشنهادی ما از یک رابطه پویا استفاده کرده‌ایم که در فواصل زمانی مشخص مقدار فرمون کارها به‌روز رسانی می‌شود و مقدار فرمون بر اساس شرایط جدید سیستم محاسبه می‌شود.

در روش پایه برای محاسبه فرمون از زمان ورود کار و اندازه کار استفاده می‌شود ولی در روش پیشنهادی علاوه بر زمان ورود و اندازه زمان مرگ هم به‌عنوان عامل جدید در نظر گرفته شده است.

در روش پایه الگوریتم کلونی مورچه هزینه‌ای که کار برای ماشین ایجاد می‌کند در نظر گرفته نشده است ولی در روش پیشنهادی هزینه اجرای کار هم به‌عنوان یک عامل مهم در نظر گرفته شده است.

۱-۱۱-منابع

- Gao, Y., Guan, H., Qi, Z., Hou, Y., Liu, L. "A multi-objective ant colony system algorithm for virtual machine placement in cloud computing," *Journal of Computer and System Sciences*, Vol. 79, No. 8, pp. 1230-1242, 2013.
- Hu, W.X., Zheng, J., Hua, X.Y., Yang, Y.Q. "A computing capability allocation algorithm for cloud computing environment," *Applied Mechanics and Materials*, pp. 2400-2406, 2013.
- Liang, Y., Rui, Q.P., Xu, J. "Computing resource allocation for enterprise information management based on cloud platform Ant Colony Optimization Algorithm," *Advanced Materials Research*, pp. 1237-1232, 2013.
- Kansal, N.J., Chana, I. "Artificial bee colony based energy-aware resource utilization technique for cloud computing," *Concurrency and Computation: Practice and Experience*, Vol. 27, No. 5, pp. 1207-1225, 2015.
- Liu, X.-F., Zhan, Z.-H., Du, K.-J., Chen, W.-N. "Energy aware virtual machine placement scheduling in cloud computing based on ant colony optimization approach," *Proceedings of the Annual Conference on Genetic and Evolutionary Computation*, pp. 41-46, 2013.
- Arianyan, E., Maleki, D., Yari, A., Arianyan, I. "Efficient resource allocation in cloud data centers through genetic algorithm," *Telecommunications (IST), Sixth International Symposium on*, pp. 570-566, 2012.
- Portaluri, G., Giordano, S., Kliazovich, D., Dorronsoro, B. "A power efficient genetic algorithm for resource allocation in cloud computing data centers," *Cloud Networking (CloudNet), IEEE 3rd International Conference on*, pp. 63-58, 2014.
- Radhakrishnan, A., Kavitha, V. "Trusted virtual machine allocation in cloud computing IaaS service," *Research Journal of Applied Sciences, Engineering and Technology*, Vol. 7, No. 14, pp. 2921-2928, 2014.
- Lin, Y.-K., Chong, C.S. "Fast GA-based project scheduling for computing resources allocation in a cloud manufacturing system," *Journal of Intelligent Manufacturing*, pp. 1-13, 2015.
- Xiong, A.-p., Xu, C.-x. "Energy efficient multiresource allocation of virtual machine based on PSO in cloud data center," *Mathematical Problems in Engineering*, Vol. 2014, No. 2014.
- Dashti, S.E., Rahmani, A.M. "Dynamic VMs placement for energy efficiency by PSO in cloud computing," *Journal of Experimental & Theoretical Artificial Intelligence*, Vol. 28, No. 1-2, pp. 97-112, 2016.
- Kumar, D., Raza, Z. "A PSO based VM resource scheduling model for cloud computing," *Computational Intelligence & Communication Technology (CICT), IEEE International Conference on*, pp. 219-213, 2015.

Topic: Review of algorithms based on load balancing and resource allocation

1- Amin Karimullah Dehkordi

(Master of Computer Engineering)

2- Dr. Golnaz Aghaei

(specialized doctorate in software systems and university assistant professor)

Abstract

In recent years, with the expansion of the use of cloud computing technology and the increase in the number of users in mobile cloud systems, the effective management of resources and load balancing has become a fundamental challenge. These challenges are especially manifested in mobile cloud environments that have limitations such as limited energy and high fluctuations in resource demand. Many reference methods in this field, including algorithms based on load balancing and resource allocation, usually face problems such as high energy consumption, long response time, and lack of proper load balancing. These issues lead to a decrease in the overall efficiency of cloud systems and an increase in costs. In this article, we discuss and compare scheduling and resource allocation algorithms with an overview of the latest approaches and present a classification of these algorithms.

Keywords: scheduling, resource allocation, load balancing