# (رایانش ابری ، چالش ها و بررسی روش‌ها)

**سجاد میر**
گروه کامپیوتر دانشگاه ملی مهارت
**فردین فلاحی**
دانشگاه ملی مهارت
**محمود فرزامی**
دانشگاه ملی مهارت

**چکیده**

رایانش ابری یکی از فناوری‌های مهم محاسباتی است که امکان ارائه خدمات و منابع مبتنی بر تقاضا از برنامه کاربر به مرکز داده ابری با استفاده از اتصال اینترنت به صورت پرداخت به میزان تقاضا را فراهم می‌کند. ایده اصلی توازن بار، بهینه‌سازی استفاده از منابع، هزینه‌ی مرکز داده و ماشین‌های مجازی، به حداکثر رساندن توان عملیاتی ، کاهش زمان پاسخ و جلوگیری از بارگذاری زیاد در ماشین‌های مجازی مختلف و همچنین سرورهای ابری است.

این مقاله به تحلیل مزایا و معایب هر روش می‌پردازد و بر اساس مطالعات انجام شده، میزان کارایی، عدالت در توزیع منابع، مقیاس‌پذیری و پیچیدگی پیاده‌سازی هر یک از این روش‌ها را مقایسه می‌کند و به بررسی چالش‌ها و مسائل توزیع بار در رایانش ابری می‌پردازد . به این ترتیب، مقاله به درک بهتر و عمیق‌تری از چگونگی عملکرد و کاربرد الگوریتم‌های توازن بار در عمل می‌انجامد، و دیدگاه‌های جدیدی را در زمان بندی وظایف برای تحقیقات آینده ارائه می‌دهد که می‌تواند به حل چالش‌های موجود و بهبود کارایی سیستم‌های ابری کمک کند.

**واژگان کلیدی: رایانش ابری ، توزیع بار ، سیستم‌های توزیع شده**

**مقدمه**

رایانش ابری یکی از الگوهای کارآمد برای محاسبات توزیع شده است. مهمترین مسأله در توازن بار در رایانش ابری، توزیع بار کاری و منابع محاسباتی در بین سرورهای مختلف است. [1] رایانش ابری یک اصطلاح کلی برای ارائه دسترسی به مجموعه‌ای از خدمات از طریق وب است . این خدمات شامل شبکه‌ها، سرورها، ذخیره‌سازی، برنامه‌های کاربردی، قدرت پردازش و سایر دارایی‌های مرتبط با فناوری اطلاعات می‌شود. [2]

موسسه ملی فناوری و استانداردها آمریکا (NIST) رایانش ابری را به عنوان یک مدل مبتنی بر پرداخت به ازای استفاده تعریف می‌کند که دسترسی به شبکه، با قابلیت استفاده آسان و بر اساس تقاضا را فراهم می‌آورد. [3]

توزیع بار به عنوان بخشی از خدمات خود، فرآیندی آسان و انعطاف‌پذیر برای نگهداری داده‌ها یا فایل‌ها و در دسترس قرار دادن آن‌ها برای کاربران در مقیاس بزرگ را ارائه می‌کند [4]

با توسعه‌ی فناوری رایانش ابری، ابعاد این حوزه روز به روز در حال گسترش است و وظایف بیشتری نیازمند مدیریت هستند . چگونگی تخصیص و زمان‌بندی این وظایف به طور مؤثر بر کارایی کلی و کیفیت خدمات رایانش ابری تأثیر می‌گذارد. بنابراین، الگوریتم زمان‌بندی وظایف در رایانش ابری به یک موضوع مهم در تحقیقات این حوزه تبدیل شده است. [3]

**۳. کارهای مرتبط :**

**تعداد زیادی از کارهای تحقیقاتی قبلاً برای توازن بار در محیط ابری پیشنهاد شده است. در این بخش، تعدادی از کارهای موجود در زمینه توازن بار مورد بررسی قرار گرفته است**

**M. Lagwal و N. Bhardwaj (2017) برای توازن بار کار با ترتیب دادن به ماشین‌های مجازی بر اساس قدرت پردازشی آن‌ها و ترتیب دادن به cloudlets بر اساس حجم آن‌ها، پیشنهاد دادند. پس از آن، لیستی از ماشین‌های مجازی با cloudlets مربوطه به یک کارگزار (broker) برای تخصیص ارسال می‌شود .کارگزار با استفاده از الگوریتم ژنتیک GA منابع را تخصیص می دهد . اما در این مقاله، مشکل اصلی این است که تمام وظایف با حجم کم به ماشین‌های مجازی اختصاص داده می‌شوند و وظایف با حجم بیشتر منتظر پایان این فرآیندها با حجم کوچک هستند [5]**

**H. A Makasarwala و P. Hazari (2016) درباره یک الگوریتم مبتنی بر ژنتیک برای توازن بار در محیط ابری بحث کردند. در این روش، اولویت یک درخواست بر اساس زمان مورد نیاز آن برای تخصیص بهتر به جمعیت اولیه در نظر گرفته می‌شود. نشان داده شده است که طول کار (حجم) بیشتر به معنای زمان‌بر بودن است. اولویت با کارهایی است که نیاز به زمان کمتری دارند . برای تحلیل و مقایسه عملکرد با سایر الگوریتم‌های پایه‌ای توازن بار در ابر، شبیه‌سازی Cloud Analystاستفاده شده است. در این مقاله، میانگین زمان makespan برایcloudlets مختلف محاسبه نشده است و همچنین ترافیک شبکه برای انتخاب مناسب ترین ماشین مجازی در نظر گرفته نشده است. [6]**

**P. Biswal ، C.K. Rath و S. S. Suar (2018) الگوریتم ژنتیکی را برای برنامه ریزی پویای وظایف با توازن بار در محیط ابری به نمایش گذاشتند. از انحراف معیار برای یافتن یک تابع fitness جدید استفاده شده است. به حداقل رساندن زمان makespanاصلی هدف الگوریتم پیشنهادی آن‌ها بود. نتایج با استفاده از سیاست برنامه‌ریزی پیشنهادی نشان داده شده است که می‌تواند زمان خالی و زمان makespan را با استفاده مناسب از پردازنده و توزیع بار به طور مناسب کاهش دهد. [7]**

O. Kaneria و R. K. Banyal (2016) پیشنهاد دادند که از منابع به‌طور کارآمد در منابع مبتنی بر ابر استفاده شود و سرعت دسترسی را با تغییر الگوریتم‌های توازن بار افزایش دهند، با اختصاص یک وظیفه به یک میزبان که بیشترین تعداد پردازنده را دارد. اختصاص cloudlets به ماشین‌های مجازی در یک مرکز داده با استفاده از یک سیاست تخصیص منابع انجام می شود. ابزار Cloud-Sim برای شبیه‌سازی استفاده شده است. [8]

A. Dave، P. Bhargesh و G. Bhatt (2016) یک بررسی در مورد تکنیک‌های مختلف بهینه‌سازی برای توازن بار را مورد بحث قرار دادند . در این مقاله، رویکردهای تکاملی و بهینه‌سازی مبتنی بر ازدحام برای سناریوهای توازن بار مورد بحث قرار گرفته است. [9]

A. Ragmani، A. E. Omri، N. Abghour، K. Moussaid و M. Rida (2016) انواع رویکردها و مطالعاتی را که برای روش‌های مختلف توازن بار در رایانش ابری استفاده شده است، پیشنهاد کردند. آن‌ها همچنین یک سبک توازن بار بهتر و یک تکنیک جدید برای رایانش ابری را توصیه کردند که زمان پاسخ بهبود یافته‌ای را ارائه داد. [10]

R. Beri و V. Behal (2015) یک بررسی درباره رایانش ابری انجام دادند که در آن مدل‌های مختلف خدمات در رایانش ابری را توضیح دادند. با کمک این تحقیق، می‌توانیم دسته‌بندی‌های مختلف خدمات ارائه شده را تجزیه و تحلیل کنیم. این مطالعه توصیف مختصری از معماری لایه‌ای ابر را ارائه داد و نحوه کار خدمات ابری را ارزیابی کرد. [11]

X. Zongyu و W. Xingxuan (2015) روشی را برای پیش‌بینی سرور کارآمد در زمینه رایانش ابری پیشنهاد دادند. آن‌ها یک الگوریتم توازن بار مبتنی بر Round Robin (RR) اصلاح‌شده را پیشنهاد کرده‌اند، و کارایی این الگوریتم پیشنهادی از طریق شبیه‌سازی آن الگوریتم سنجیده شده است. نتایج نشان می‌دهد که الگوریتم RR اصلاح‌شده، بار را به شدت کاهش داده و اختلاف بار را کم می‌کند. [12]

جدول ۱ : مقایسه روش های کارهای مرتبط

| مقاله | روش | مزایا | معایب |
|---|---|---|---|
| M. Lagwal و N. Bhardwaj 2017 | این الگوریتم از الگوریتم ژنتیک (GA) برای مرتب‌سازی ماشین‌های مجازی بر اساس قدرت پردازشی و مرتب‌سازی وظایف بر اساس حجم آنها استفاده می‌کند. | ساده برای پیاده‌سازی، کارآمد | وظایف با حجم بزرگ منتظر تکمیل وظایف کوچک می‌مانند که منجر به تأخیر می‌شود. |

| | | | |
|---|---|---|---|
| P. و H. A Makasarwala Hazari 2016 | از الگوریتم ژنتیک برای اولویت‌بندی وظایف بر اساس زمان مورد نیاز آنها برای تخصیص استفاده می‌کند. | اولویت‌دهی به وظایف با حجم کمتر و کاهش زمان کل اجرای وظایف. | میانگین زمان makespan را محاسبه نمی‌کند |
| S. S. و P. Biswal،C.K. Rath Suar 2018 | از الگوریتم ژنتیک با استفاده از انحراف معیار برای یافتن تابع fitness جدید برای به حداقل رساندن زمان makespan استفاده می‌کند. | کاهش زمان makespan و زمان انتظار ، استفاده بهینه از پردازنده و توزیع مناسب بار | پیچیدگی بالا |
| O. Kaneria و R. K. Banyal 2016 | تخصیص وظایف به میزبان با بیشترین تعداد پردازنده | استفاده کارآمد از منابع و افزایش سرعت دسترسی | ترافیک شبکه را در نظر نمی‌گیرد |
| G. و P. Bhargesh ،A. Dave Bhatt 2016 | مروری بر تکنیک‌های مختلف بهینه‌سازی برای توازن بار ارائه می‌دهد. | بررسی روش‌های بهینه‌سازی توازن بار | ارائه راه‌حل جدید ارائه نمی‌دهد |
| ،A. E. Omri ،A. Ragmani ،K. Moussaid ،N. Abghour M. Rida 2016 | پیشنهاد روش‌های توازن بار مختلف و توصیه تکنیک جدید برای بهبود زمان پاسخ | ارائه رویکرد جدید برای توازن بار | جزئیات پیاده‌سازی ارائه نشده است |
| R. Beri و V. Behal 2015 | مدل‌های مختلف خدمات ابری را بررسی می‌کند. | ارائه توضیح مختصر و مفید از خدمات ابری | بیشتر بر روی تئوری تمرکز دارد تا کاربرد عملی |
| X. Zongyu و W. Xingxuan 2015 | الگوریتم RR اصلاح‌شده را برای کاهش بار و اختلاف بار ارائه می‌دهد. | کاهش بار و اختلاف بار | فقط الگوریتم RR را اصلاح می‌کند |

جدول ۲ : مقایسه روش های زمانبندی وظایف

| معیار | Lagwalو Bhardwaj 2017 | Makasarwala و Hazari 2016 | ،Rath Biswal و Suar 2018 | Kaneria و Banyal 2016 | ،Dave Bhargesh و Bhatt 2016 | ،Ragmani ،Omri ،Abghour ،Moussaid و Rida 2016 | Beri و Behal 2015 | Zongyu و Xingxuan 2015 |
|---|---|---|---|---|---|---|---|---|
| کارایی | متوسط | بالا | بالا | بالا | متوسط | بالا | متوسط | بالا |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| عدالت در توزیع منابع | بالا | بالا | بالا | متوسط | – | بالا | – | بالا |
| مقیاس‌پذیری | خوب | خوب | خوب | خوب | – | خوب | – | خوب |
| پیچیدگی پیاده سازی | بالا | بالا | بالا | متوسط | متوسط | بالا | متوسط | متوسط |

**۴. چالش‌ها و مسائل توزیع بار**

**با وجود گسترش روزافزون استفاده از رایانش ابری، تحقیقات در این زمینه همچنان در مراحل اولیه قرار دارد [13]. بنابراین، پیش از شرح الگوریتم‌های توزیع بار برای رایانش ابری، شناسایی چالش‌ها و مسائلی که بر عملکرد این الگوریتم‌ها تأثیر می‌گذارند، ضروری است [14].**

**۱.۴. تأمین خودکار سرویس (Automated Service Provisioning) :**

**یکی از اجزای اصلی رایانش ابری، قابلیت ارتجاعی (Elasticity) است که به تخصیص و آزادسازی منابع به صورت خودکار منجر می‌شود. چالش اصلی این است که چگونه می‌توان همزمان از قابلیت ارتجاعی ابر برای بهینه‌سازی مصرف منابع استفاده کرد و در عین حال، عملکرد سیستم‌های سنتی را حفظ نمود [13]**

**۲.۴. انتقال پذیری ماشین مجازی (Virtual Machine Migration) :**

**ایده‌ی این مفهوم این است که یک ماشین مجازی را به صورت مجموعه‌ای از فایل‌ها در نظر بگیریم. با انتقال ماشین‌های مجازی بین گره‌های مختلف به صورت مؤثر، می‌توان بار روی گره‌های پرکار را کاهش داد. هدف نهایی توزیع تمام انواع بار در مرکز داده است. چالش اصلی در هنگام توزیع پویای بار با استفاده از انتقال پذیری ماشین مجازی، از معایب و مشکلات احتمالی سیستم رایانش ابری جلوگیری شود**

**۳.۴. مدیریت انرژی (Energy Management) :**

**مدیریت انرژی یکی از نکات کلیدی است که به کاربران امکان می‌دهد از منابع موجود در مراکز داده‌ی جهانی استفاده کنند. این موضوع منجر به صرفه‌جویی درمقیاس (economies of scale) می‌شود و یک مزیت عمده برای رایانش ابری به شمار می‌رود. با این حال، یک سوال اساسی مطرح می‌شود: چگونه می‌توان با استفاده از بخشی از ظرفیت یک مرکز داده، همچنان عملکرد خوبی را تضمین کرد ؟**

**۴.۴. مدیریت داده‌های ذخیره‌سازی شده (Stored Data Management) :**

**یکی دیگر از نیازهای کلیدی در توزیع بار ابری، مدیریت داده‌های ذخیره‌سازی شده است. چالش اصلی این است که چگونه داده‌ها را در سیستم ابری توزیع کنیم تا هم از مناسب‌ترین فضای ذخیره‌سازی استفاده شود و هم دسترسی سریع به آن‌ها امکان‌پذیر باشد.**

**۵.۴. ظهور مراکز داده کوچک برای رایانش ابری(Emergence of Small Data Centers for Cloud Computing) :**

**مراکز داده کوچک می‌توانند مزایای بیشتری داشته باشند زیرا انرژی کمتری مصرف می‌کنند و هزینه کمتری نسبت به مراکز داده بزرگ‌تر دارند. توزیع بار به عنوان یک مسئله در مقیاس جهانی مطرح است که برای تضمین زمان پاسخ مناسب با بهره‌وری و توزیع بهینه منابع، اهمیت ویژه‌ای دارد.**

**۶.۴. توزیع مکانی گره‌های ابری (Spatial Distribution of Cloud Nodes) :**

**بسیاری از الگوریتم‌های توزیع بار برای سناریوهایی طراحی شده‌اند که گره‌های ابری (مانند ماشین‌های مجازی) در یک مکان واحد قرار دارند و تأخیر ارتباط بین آن‌ها ناچیز است [14]. با این حال ، طراحی یک الگوریتم کارآمد توزیع بار که بتواند به درستی برای گره‌های توزیع‌شده‌ی جغرافیایی (spatially distributed) فرمول‌بندی شود، همچنان یک چالش است [15].**

**۷.۴. ذخیره‌سازی و تکثیر (Storage and Replication) :**

**الگوریتم تکثیر کامل برای استفاده بهینه از فضای ذخیره‌سازی در یک سیستم مناسب نیست .دلیل این امر آن است که در این روش، یک نسخه کامل از داده‌ها در تمام گره‌های تکثیر شده نگهداری می‌شود .الگوریتم‌های تکثیر کامل به دلیل نیاز به فضای ذخیره‌سازی زیاد، منجر به هزینه‌های غیرمنطقی می‌شوند [14] .**

**۹.۴. پیچیدگی الگوریتم (Algorithm Complexity) :**

**الگوریتم‌های توزیع بار ترجیح بر این است که از نظر عملیات و پیاده‌سازی (implementation) ساده باشند .پیچیدگی منفی در پیاده‌سازی منجر به فرایندی بسیار پیچیده می‌شود .علاوه بر این، برای نظارت و کنترل پیاده‌سازی، الگوریتم‌ها به برقراری ارتباط بیشتر، اطلاعات بیشتر و زمان تأخیر بیشتری نیاز دارند که این عوامل می‌توانند باعث ایجاد گلوگاه (bottleneck)و در نهایت کاهش کارایی شوند [14].**

**۱۰.۴. کنترل نقطه شکست (Point of Failure Controlling) :**

**برخی از الگوریتم‌های توزیع بار، به ویژه الگوریتم‌های متمرکز(Centralized Algorithms) ، مکانیزم‌های کارامدی را برای پردازش توزیع بار با یک الگوی خاص ارائه می‌دهند .با این حال، چالش این الگوریتم‌ها وجود تنها یک کنترلر برای کل سیستم است .در چنین شرایطی، اگر کنترلر از کار بیفتد، کل سیستم با مشکل مواجه خواهد شد . [14]**

**۵. بر اساس دسترسی، رایانش ابری به دسته‌های زیر تقسیم می‌شود:**

**۱. ابر عمومی (Public Cloud) : در این نوع ، یک شرکت ثالث، دارایی‌های مشترک، سیستم و ظرفیت را از طریق وب در اختیار کاربران قرار می‌دهد . مثال : Salesforce.com**

**معماری ابر عمومی به مدل‌های خدماتی زیر تقسیم می‌شود:**

**۱.۱. نرم‌افزار به عنوان سرویس (SaaS) : SaaS در بالاترین سطح قرار دارد .این مدل با ارائه نرم‌افزار ابری فروشنده به کاربران نهایی، به رایانش ابری کمک می‌کند.**

**۲.۱. پلتفرم به عنوان سرویس (PaaS) : PaaS در سطح پایین‌تری نسبت به SaaS قرار دارد . PaaS با ارائه یک پلتفرم ابری به کاربر، امکان توسعه، مدیریت و ارائه خدمات ابری را برای او فراهم می‌کند.**

**۱.۳. زیرساخت به عنوان سرویس (IaaS) : IaaS در پایین‌ترین سطح قرار دارد .این مدل با ارائه دسترسی به منابع محاسباتی مانند ماشین‌های مجازی، سیستم‌ها، سرورها و فضای ذخیره‌سازی پایگاه داده به کاربران، به رایانش ابری کمک می‌کند.**

**۲. ابر خصوصی (Private Cloud) :**

**برخلاف ابر عمومی، ابر خصوصی توسط یک سازمان یا شرکت خاص نگهداری می‌شود. برخلاف ابر عمومی، ابر خصوصی به یک سازمان اختصاص داده شده است . به عنوان مثال، یک سازمان دولتی.**

**۳. ابر ترکیبی(Hybrid Cloud) :**

**ابر ترکیبی ترکیبی از ابر خصوصی درون‌سازمانی و سرویس‌های ابر عمومی شخص ثالث به همراه اتصالی بین این دو است . ابر ترکیبی امکان انتقال حجم کاری بین ابر خصوصی و عمومی را فراهم می‌کند که منجر به انعطاف‌پذیری و تطبیق‌پذیری برجسته‌تر می‌شود.**

**امروزه تقاضا برای منابع رایانش ابری در حال افزایش است و به این ترتیب تعداد مشتریان نهایی که به این سرویس‌ها دسترسی پیدا می‌کنند، رو به رشد است .به همین دلیل، نیاز به تنظیم بار سرورهای توزیع‌شده مطرح می‌شود.**

**۶. توزیع بار (Load Balancing) :**

**توزیع بار به فرآیند توزیع یکنواخت‌تر ترافیک شبکه یا برنامه در میان یک مجموعه سرور اشاره دارد .توزیع بار، عملکرد و قابلیت اطمینان برنامه‌های کاربردی را بهبود می‌بخشد.**

**توزیع بار با ارائه یک راه‌حل مؤثر برای مسائل مختلف موجود در راه‌اندازی و استفاده از محیط رایانش ابری، به رایانش ابری کمک می‌کند. توزیع بار بر تخصیص منابع و زمان‌بندی وظایف در محیط توزیع‌شده تمرکز دارد. همچنین به موارد زیر کمک می‌کند:**

**۱.۶. جلوگیری از گلوگاه‌های ترافیکی شبکه (avoiding bottlenecks issue in network traffic) :توزیع بار به توزیع یکنواخت ترافیک در بین سرورهای موجود کمک می‌کند و از بارگذاری بیش از حد روی یک سرور خاص جلوگیری می‌نماید. این امر مانع از بروز گلوگاه‌های ترافیکی می‌شود که می‌تواند منجر به کند شدن عملکرد کل سیستم شود.**

**۲.۶. کاهش زمان پاسخ (reducing response time) : با توزیع یکنواخت ترافیک در میان سرور ها ، کاربران زمان کمتری را برای دریافت پاسخ به درخواست‌های خود صرف می‌کنند. این امر منجر به بهبود کلی تجربه کاربری می‌شود.**

**۳.۶. کاهش هزینه کلی (lessening overall cost) : توزیع بار به استفاده بهینه‌تر از منابع محاسباتی کمک می‌کند، که منجر به کاهش هزینه‌های کلی می‌شود.**

**۴.۶. افزایش انعطاف‌پذیری در برابر تحمل خطاهای غیربحرانی**

**(enhancing adaptation to non-critical failure capacity) : در صورت خرابی یک سرور، توزیع بار به صورت خودکار ترافیک را به سرورهای باقیمانده هدایت می‌کند و در نتیجه، باعث افزایش تحمل خطا و در دسترس‌بودن سرویس را حفظ می‌کند.**

**۵.۶. درک مسائل امنیتی (comprehending security issue) : توزیع بار می‌تواند به شناسایی و رفع مسائل امنیتی بالقوه کمک کند. با توزیع ترافیک به صورت ایمن بر روی سرورهای مختلف، یک نقطه ورود واحد برای مهاجمان از بین می‌رود. [16]**

**۷. دیدگاه‌های جدید برای تحقیقات آینده در زمان‌بندی وظایف رایانش ابری**

**زمان‌بندی وظایف در رایانش ابری یک حوزه تحقیقاتی فعال است که به دنبال روش‌هایی برای تخصیص کارآمد منابع محاسباتی به وظایف برای بهینه‌سازی عملکرد و کارایی است .با توجه به پیچیدگی و پویایی روزافزون محیط‌های ابری، نیاز به الگوریتم‌های زمان‌بندی جدید و نوآورانه که بتوانند با این چالش‌ها مقابله کنند، روز به روز بیشتر می‌شود.**

**در اینجا چند دیدگاه جدید برای تحقیقات آینده در زمان‌بندی وظایف رایانش ابری ارائه می‌شود:**

۱ . یادگیری ماشین و هوش مصنوعی:

۱.۱. استفاده از یادگیری ماشین برای یادگیری الگوهای بار کاری و پیش‌بینی تقاضا برای منابع محاسباتی.

۲.۱. استفاده از هوش مصنوعی برای توسعه الگوریتم‌های زمان‌بندی تطبیقی که می‌توانند با شرایط متغیر محیط ابری سازگار شوند.

۳.۱. ادغام یادگیری تقویتی عمیق برای بهینه‌سازی زمان‌بندی وظایف در زمان واقعی. [17]

۲ . زمان‌بندی چندوظیفه‌ای:

۱.۲. توسعه الگوریتم‌های زمان‌بندی که می‌توانند به طور همزمان چندین وظیفه را با نیازها و الزامات مختلف مدیریت کنند.

۲.۲. در نظر گرفتن وابستگی‌های بین وظایف برای بهینه‌سازی جریان کار.

۳.۲. استفاده از برنامه‌ریزی ترکیبی برای حل مسائل زمان‌بندی پیچیده چندوظیفه‌ای. [18]

۳. زمان‌بندی آگاه از کیفیت خدمات (QoS) :

۱.۳. در نظر گرفتن الزامات QoS مانند زمان پاسخ، پهنای باند و نرخ خطا در هنگام زمان‌بندی وظایف.

۲.۳. توسعه الگوریتم‌های زمان‌بندی که می‌توانند تعادل بین کارایی و الزامات QoS را برقرار کنند.

۳.۳. استفاده از تکنیک‌های کنترل صف برای مدیریت ترافیک و تضمین کیفیت خدمات برای وظایف حساس به زمان. [19]

۴ . زمان‌بندی در محیط‌های ابری هیبریدی و چند ابری:

۱.۴. توسعه الگوریتم‌های زمان‌بندی که می‌توانند وظایف را در چندین ابر به طور کارآمد توزیع کنند.

۲.۴. در نظر گرفتن هزینه محاسباتی و تاخیر شبکه در هنگام زمان‌بندی وظایف در محیط‌های ابری هیبریدی.

۳.۴. استفاده از تکنیک‌های فدرال برای مدیریت منابع و زمان‌بندی وظایف در چندین ابر. [20]

۵ . زمان‌بندی در محیط‌های لبه:

۱.۵. توسعه الگوریتم‌های زمان‌بندی که می‌توانند با محدودیت‌های منابع و تاخیر شبکه در محیط‌های لبه سازگار شوند.

۲.۵. استفاده از تکنیک‌های تخلیه محاسباتی برای انتقال وظایف به ابر برای پردازش.

۳.۵. توسعه الگوریتم‌های زمان‌بندی پراکنده برای هماهنگی وظایف در چندین دستگاه لبه. [21]

۸. الگوریتم‌های موجود توزیع بار

برای دستیابی به توان عملیاتی (throughput) کارآمد، کاهش زمان پاسخگویی و جلوگیری از اضافه بار بر روی یک منبع خاص، از الگوریتم‌های توزیع بار (ایستا و پویا) استفاده می‌شود [22]. این الگوریتم‌ها در جدولی به همراه مزایا و معایب آن‌ها مقایسه شده‌اند: [4]

جدول ۳ : مقایسه الگوریتم توازن بار موجود

| الگوریتم ها | ایستا / پویا | توضیح | مزایا | معایب |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Round Robin and Randomized | ایستا | ۱. در این الگوریتم توزیع بار، فرآیندها به طور مساوی بین تمامی پردازنده‌ها تقسیم می‌شوند.<br>۲. ترتیب تخصیص فرآیند به صورت محلی بر روی هر پردازنده نگهداری می‌گردد.<br>۳. با استفاده از این الگوریتم، درخواست‌های کاربر به صورت دوره‌ای پردازش می‌شوند. | این الگوریتم در شرایطی که تعداد فرآیندها از تعداد پردازنده‌ها بیشتر باشد، عملکرد خوبی از خود نشان می‌دهد.<br>این الگوریتم نیازي به برقراري ارتباط بین فرآیندها ندارد. این امر پیاده‌سازي آن را ساده‌تر مي‌کند، زیرا نیازي به هماهنگي بین فرآیندها براي برنامه‌ریزي اجرا نیست. | در تخصیص دوره‌ای (Round Robin)، انتظار دستیابی به عملکرد بهینه وجود ندارد. |
| Central Manager | ایستا | ۱. پردازنده مرکزیمسئول انتخاب میزبان (Host) برای تمامی فرآیندهای جدید است.<br>۲. در هنگام ایجاد فرآیند، پردازنده با کمترین بار بر اساس کل بار سیستم، به عنوان میزبان انتخاب می‌شود. | زمان‌بندی بار (Load Scheduler) بر اساس اطلاعات بار سیستم، تصمیمات توزیع بار را اتخاذ می‌کند. | نیاز به برقراری ارتباط سطح بالایی بین فرآیندها که منجر به ایجاد گلوگاه می‌شود. |
| Threshold | ایستا | ۱. فرآیندها بلافاصله پس از ایجاد، بدون تأخیر به میزبان‌ها اختصاص داده می‌شوند.<br>۲. اختصاص میزبان برای فرآیندهای جدید به صورت منطقه‌ای (regionally) انجام می‌شود و نیازی به ارسال پیام‌های از راه دور نیست. | الگوریتم آستانه (Threshold) ارتباط بین فرآیندی کم دارد تخصیص‌های متعددی به صورت محلی برای فرآیندها انجام می‌شود | هنگامی که تمامی فرآیندهای از راه دور با اضافه بار مواجه شوند، همه فرآیندها به صورت محلی اختصاص داده می‌شوند. |

| | | | | |
|---|---|---|---|---|
| Min-Min | ایستا | ۱. این الگوریتم برای همه وظایف(Tasks) ، کمترین زمان تکمیل (Minimum Completion Time)را جستجو می‌کند.<br>۲. از بین زمان‌های حداقل به دست آمده، کمترین مقدار انتخاب می‌شود که نشان‌دهنده‌ی سریع ترین زمان تکمیل در بین تمامی وظایف روی هر نوع منبعی در سیستم است.<br>۳. بر اساس این حداقل زمان، وظیفه به ماشین (Machine) مناسب در سیستم اختصاص داده می‌شود. | با تعداد منابع بهینه ، عملکرد خوبی از خود نشان می‌دهد. | می‌تواند منجر به گرسنگی (Starvation)شود. |
| Max-Min | ایستا | الگوریتم توزیع بار (Max-Min) شباهت زیادی به الگوریتم (min-min) دارد، اما یک تفاوت کلیدی وجود دارد:<br>پس از به دست آوردن حداقل زمان‌های اجرای وظایف روی هر منبع، مقدار حداکثر انتخاب می‌شود. به عبارت دیگر، این مقدار نشان‌دهنده‌ی طولانی‌ترین زمان اجرای یک وظیفه خاص بر روی هر نوع منبع در سیستم است.<br>در نهایت، وظیفه به ماشینی اختصاص داده می‌شود که بتواند آن را با حداقل زمان کل (با در نظر گرفتن زمان‌های اجرا روی سایر منابع) تکمیل کند. | با تعداد منابع بهینه ، عملکرد خوبی از خود نشان می‌دهد. | می‌تواند منجر به گرسنگی (Starvation)شود. |
| Honey Bee Foraging Behavior | پویا | با استفاده از اقدامات ساده در سطح سرورهای محلی، به توزیع بار در سطح جهانی دست می‌یابد. | با افزایش تنوع سیستم، عملکرد آن نیز بهبود می‌یابد. | با افزایش اندازه سیستم، توان عملیاتی (throughput) آن افزایش نمی‌یابد. |
| Biased Random sampling | پویا | با استفاده از نمونه‌برداری تصادفی از دامنه سیستم، توزیع بار را در بین تمامی گره‌های سیستم برقرار می‌کند. | عملکرد آن زمانی عالی است که تعداد زیادی از منابع مشابه در اختیار داشته باشد. | با افزایش تنوع جمعیت منابع موجود در سیستم عملکرد آن کاهش می‌یابد. |

| | | | | |
|---|---|---|---|---|
| Active Clustering | پویا | تخصیص وظایف را با اتصال سرویس‌های مشابه از طریق سیم‌کشی مجدد محلی بهینه می‌کند. | ۱. عملکرد خوبی با منابع پرکاربرد دارد.<br>۲. با استفاده از افزایش منابع سیستم، توان عملیاتی (throughput)را نیز افزایش می‌دهد. | با افزایش تنوع سیستم، عملکرد آن کاهش می‌یابد. |
| ACCLB(Ant Colony and Complex network Theory) | پویا | این روش از ویژگی‌های جهان کوچک و توزیع بدون مقیاس شبکه‌های پیچیده برای دستیابی به توزیع بار مناسب استفاده می‌کند. | ۱. کنترل ناهمگنی<br>۲. سازگار با محیطهای پویا<br>۳. مقاومت در برابر خطا<br>٤. قابلیت مقیاس‌پذیری خوب | ۱. تنها در شبکه‌های پیچیده به کار می‌رود<br>۲. مقیاس‌پذیری و عملکرد پایین |
| Compare and Balance | پویا | ۱. بر اساس نمونه‌برداری<br>۲. از مهاجرت زنده تطبیقی ماشین‌های مجازی (VMs) استفاده می‌کند | ۱. توازن بار بین سرورها را برقرار می‌کند<br>۲. انتقال ماشین‌های مجازی (VMs) از میزبان‌های فیزیکی پرهزینه به میزبان‌های کم‌هزینه‌تر سیستم را تضمین می‌کند | بسته به فرض وجود حافظه کافی در هر میزبان فیزیکی |
| Vector Dot | پویا | از حاصل ضرب نقطه‌ای برای تمایز گره‌ها بر اساس نیازمندی‌های آیتم استفاده می‌کند. | ۱. نظارت بر محدودیت‌های منابع چندبعدی و سلسله مراتبی<br>۲. از بین بردن اضافه بار روی سرورها، سوئیچ‌ها و حافظه | تنها برای شبکه‌های پیچیده قابل استفاده است |
| GA Algorithm | پویا | ۱. الگوریتم ژنتیک (Genetic Algorithm)یک روش جستجوی اکتشافی است که بر اساس الگوریتم تکاملی با انتخاب طبیعی عمل می‌کند.<br>۲. الگوریتم ژنتیک عمدتاً بر سه مرحله تمرکز دارد: انتخاب (selection)، ترکیب (crossover) و جهش(mutation) | ۱. از رویکرد توزیع بار متمرکز استفاده می‌کند<br>۲. این الگوریتم برنامه‌های زمان‌بندی کارآمدی ارائه می‌دهد | این روش بر اساس رویکرد توزیع شده برای توزیع بار عمل نمیکند |

۹. نتیجه گیری :

توزیع بار در رایانش ابری، نقشی حیاتی در ارتقای کارایی، پویایی و صرفه‌جویی اقتصادی این سیستم‌ها ایفا می‌کند. با وجود چالش‌های موجود، الگوریتم‌های مختلفی برای حل این مسئله ارائه شده‌اند که هر کدام مزایا و معایب خاص خود را

دارند. انتخاب الگوریتم مناسب، به عوامل مختلفی از جمله نوع وظایف، ترافیک شبکه، ساختار سیستم و نیازهای کاربر بستگی دارد.

منابع :

[1] Modified Genetic Based Algorithm for Load Balancing in Cloud Computing
[2] HEURISTIC LOAD BALANCING ALGORITHMS IN VULNERABLE CLOUD COMPUTING ENVIRONMENT
[3] Cloud Computing Task Scheduling Algorithm Based On Improved Genetic Algorithm
[4] Load Balancing in Cloud Computing:Challenges & Issues
M. Lagwal and N. Bhardwaj, “Load balancing in cloud computing using genetic algorithm,” 2017 [5] International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, 2017, pp. 560–565.
[6] H. A Makasarwala and P. Hazari, “Using Genetic Algorithm for Load Balancing in Cloud Computing,” 2016 IEEE International Conference on Electronics, Computers and Artificial Intelligence, Romania, 2016, pp.1–6.
[7] C.K. Rath, P. Biswal and S.S. Suar, “Dynamic Task Scheduling with Load Balancing using Genetic Algorithm,” 2018 International Conference on Information Technology (ICIT), Bhubaneswar, 2018, pp. 91–95.
[8] O. Kaneria and R. K. Banyal, “Analysis and improvement of load balancing in Cloud Computing,” 2016 International Conference on ICT in Business Industry & Government (ICTBIG), Indore, 2016, pp. 1–5, DOI: 10.1109/ICTBIG.2016.7892711.
[9] A. Dave, P. Bhargesh and G. Bhatt, “Load balancing in cloud computing using optimization techniques: A study,” 2016 International Conference on Communication and Electronics Systems (ICCES), Coimbatore, 2016, pp. 1–6. 10.1109/CESYS.2016.7889883.
[10] A. Ragmani, A. E. Omri, N. Abghour, K. Moussaid, M. Rida, “A Performed Load Balancing Algorithm for Public Cloud Computing Using Ant Colony Optimization,” 2016 2nd International Conference on Cloud Computing Technologies and Applications, Morocco, 2016, pp. 221–228.
[11] R. Beri, V. Behal, “Cloud Computing: A Survey on Cloud Computing,” International Journal of Computer Applications. 2015, vol. III, no. 16.
[12] X. Zongyu and W. Xingxuan, “A predictive modified round-robin scheduling algorithm for web server clusters,” 2015 34th Chinese Control Conference (CCC), Hangzhou, 2015, pp. 5804–5808.
[13] Katoch, Swati, and J. Thakur. "Load Balancing Algoritms in Cloud Computing Environment: A Review", International Journal on Recent and Innovation Trends in Computing and Communication,Vol 2, Aug 2014
[14] Karuna G.Bakde, B .M. Patil , “Survey of techniques and challenges for load balancing in public cloud”, International Journal of Technical Research and Applications, e-ISSN: 2320-8163, Volume 4, Issue 2 (March-April, 2016), pp.279-290
[15] Buyya, Rajkumar, R. Ranjan, and R. N. Calheiros. "Intercloud: Utilityoriented federation of cloud computing environments for scaling of application services." International Conference on Algorithms and Architectures for Parallel Processing. Springer Berlin Heidelberg, 2010.
[16] HEURISTIC LOAD BALANCING ALGORITHMS IN VULNERABLE CLOUD COMPUTING ENVIRONMENT
[17] Heydari, M., & Arabnia, H. R. (2016). Machine Learning and Artificial Intelligence: Proceedings of the International Conference on Machine Learning and Computing. Springer.
[18] Dastgheibi, S., Abawajy, J., & Khosravi, A. (2016). Multi-task Scheduling in Cloud Computing Environments. Journal of Parallel and Distributed Computing, 95, 101-113

**[19] Singh, S., Chana, I., & Buyya, R. (2018). QoS-aware Scheduling of Cloud Workflows: A taxonomy and survey. ACM Computing Surveys (CSUR), 50(1), 1-41**

**[20] Srirama, S. N., Ostovar, A., & Buyya, R. (2019). A Taxonomy and Survey of Distributed Computing Systems for Hybrid and Multi-Cloud Environments. ACM Computing Surveys (CSUR), 51(6), 1-39**

**[21] Verma, P., Simmhan, Y., & Buyya, R. (2020). Enhancing Performance of Edge Computing Environments through Efficient Task Scheduling and Resource Management. Future Generation Computer Systems, 102, 230-244**

**[22] Gupta, Ruhi. "Review on existing load balancing techniques of cloud computing." International Journal of Advanced Research in Computer Science and Software Engineering 4.2 (2014): 168-71.**