الگوریتم های زمان بندی وظایف در محیط ابری

بهروز رحیم زاده، حمید طباطبایی[۲]

۱- گروه مهندسی کامپیوتر ، دانشکده مهندسی، واحد مشهد ، دانشگاه آزاد اسلامی ، مشهد ، ایران

۲- استادیار گروه مهندسی کامپیوتر، دانشگاه آزاد، مشهد، ایران

نویسنده مسئول:**behrooz.rahimzadeh@mshdiau.ac.ir**

خلاصه

رایانش ابری، یک نیاز مبتنی بر کلاینت است و منابع بسیاری را فراهم می کند که هدف آن به اشتراک گذاری آن به عنوان یک سرویس از طریق اینترنت است. برای استفاده بهینه، منابع رایانش ابری مانند ذخیره‌سازی، برنامه‌ها و سایر سرویس‌ها نیاز به مدیریت و زمان‌بندی این خدمات دارند. ایده اصلی در پس برنامه ریزی، به حداقل رساندن زمان تلفات، حجم کار و به حداکثر رساندن توان عملیاتی است. بنابراین، کار برنامه ریزی برای دستیابی به دقت و صحت در تکمیل کار ضروری است. شکل پیشرفته محاسبات مشتری-سرور، خوشه و شبکه به عنوان محاسبات ابری نامیده می شود. کاربران ابری می توانند به صورت مجازی از طریق اینترنت به منابع دسترسی داشته باشند. وظیفه ارسال شده توسط کاربران ابری مسئول کارایی و عملکرد خدمات رایانش ابری است. یکی از ضروری ترین عواملی که با به حداکثر رساندن استفاده از منابع باعث افزایش کارایی و عملکرد محیط ابری می شود، زمان بندی وظایف نامیده می شود. در این مقاله مروری، تعدادی از الگوریتم های زمان بندی وظایف مختلف در محیط محاسبات ابری مورد استفاده محققان ارائه می دهد و مقایسه چندین تکنیک زمان‌بندی کار را با توجه به پارامترهای متعددی مانند زمان پاسخ، تعادل بار، زمان اجرا و زمان انجام کار برای یافتن بهترین و کارآمد الگوریتم زمان‌بندی کار تحت این پارامترها انجام می شود.

کلمات کلیدی: **رایانش ابری، ابر، محیط ابری، الگوریتم زمانبندی، زمانبندی وظایف**

۱. مقدمه

در چند سال گذشته، رایانش ابری به عنوان یک پلت فرم محاسباتی با کارایی بالا برای افراد و سازمان ها در نظر گرفته شده است. ارائه دهندگان خدمات ابری (CSP) در حال راه اندازی مراکز داده با منابع محاسباتی با کارایی بالا هستند تا نیازهای کاربران ابری را برآورده سازند.[۲]

رایانش ابری به یک پلتفرم جذاب و محبوب برای افراد و سازمان‌ها تبدیل شده است که راه‌حلی برای اجرا و ذخیره‌سازی برنامه‌های کاربردی بزرگ ارائه می‌دهد.[۲]

پردازش ابری امکانات زیادی را برای دسترسی به داده ها در هر نقطه از ابر فراهم می کند. سازمان‌ها، کسب‌وکار خود را به سمت محاسبات ابری سوق می‌دهند؛ زیرا رایانش ابری منابع را در مقادیر زیاد فراهم می‌کند و میتوانند آزادانه از منابع استفاده می‌کنند. رایانش ابری، مدلی است که دسترسی آسان به منابع موجود را برای کاربران ابری به درخواست آنها فراهم می کند. محبوبیت ابر به عملکرد آن، مدیریت منابع و زمان‌بندی بهینه کار بستگی دارد. زمانبندی کار را می توان به عنوان انتخاب مناسب ترین منابع برای اجرای کار تعریف کرد. همچنین می‌توان آن را به‌عنوان درخواست‌های کاربر به سرورهای مختلف ارسال کرد و پاسخ دهی به این درخواستها نیز در بازه زمانی مورد نیاز انجام می‌شوند [۳].

اخیراً رایانش ابری نقش مهمی در بسیاری از سازمان ها ایفا کرده است. رایانش ابری شروع شد تا نیازهای کاربران را برای دسترسی به محاسبات منابع فراهم کند یا کاربران را قادر سازد تا خدمات ابری را همانطور که در مفهوم به اشتراک گذاری منابع بر اساس تقاضا از طریق برنامه های کاربردی بسیار مبتنی بر اینترنت لازم است خریداری کنند. رایانش ابری همچنین می تواند بسیاری از خدمات مختلف را بسته به سرویس و پلت فرم های کاری مورد نیاز کاربران پردازش کند [۱]. رایانش ابری ترکیبی از محاسبات توزیع‌شده و موازی با اشتراک منابعی مانند نرم‌افزار و سخت‌افزار است که به‌عنوان «پرداخت هزینه‌ها» استفاده می‌شود. برای استفاده از آن، کاربران نیازی به خرید پلتفرم یا نرم افزار ندارند. آنها فقط یک اتصال اینترنتی برای دسترسی و پرداخت هزینه استفاده خود دارند.

رایانش ابری یک سیستم محاسباتی توزیع شده است که نرم افزار، پردازنده، حافظه، ذخیره سازی و سایر منابع محاسباتی را ارائه می دهد. این سرویس، خدمات درخواستی را به عنوان یک سرویس پرداخت به ازای استفاده از طریق اینترنت ارائه می دهد. رایانش ابری، برای ساخت و راه اندازی محیط های رایانش ابری فناوری های مجازی سازی استفاده می شود و اجازه می دهد تا یک برنامه کاربردی یا منبع فیزیکی را بین چندین مشتری به اشتراک بگذارید و تعادل بار را می توان با مجازی سازی مدیریت کرد [۴].

معمولا مدل محاسباتی ابر مبتنی بر فناوری اطلاعات بر سهولت و سرعت تخصیص منابع فناوری اطلاعات، رهایی کاربران نهایی از زیرساخت‌های فناوری اطلاعات و مشکلات مکانی تمرکز دارد. محیط های ابری، ارائه دهندگان خدمات و اینترنت هستند. ارائه دهنده خدمات، مسئول نرم افزار، شبکه و امکانات مورد استفاده برای ایجاد سرویس هستند. ارائه دهندگان اینترنت نیز مشتریان یا مصرف کنندگان ابر نامیده می شوند [۵]. سیستم عامل های ابری (COS) به دلیل رشد سریع فناوری ابر معمولاً برای مشتریان فردی و تجاری استفاده می شود. COS یک سیستم عامل بر روی سرور است که نرم افزار کاربردی را مستقر می کند. کاربران مجبور نیستند برنامه را از طریق یک مرورگر وب از رایانه شخصی نصب کنند و به آن متصل شوند. ارائه‌دهنده ابر، مسئول ذخیره و مدیریت داده‌های ذخیره‌سازی ابری، محافظت، ارتقاء نرم‌افزار، به حداقل رساندن هزینه‌های استفاده از نرم‌افزار و ارتقای تجربه کاربری هستند [۶]. الگوریتم زمان‌بندی وظایف، وظایف کاربر را به منابع ابری اختصاص می‌دهد تا میزان استفاده را به حداکثر برساند، طول زمان عمل را کاهش دهد و زیرساخت ابر را متعادل کند تا از بارگذاری بیش از حد فعالیت‌ها جلوگیری شود.

۲. انواع ابر

- ابرهای عمومی: کاربران را قادر می سازد تا از طریق رابط های مرورگر وب از ابر استفاده کنند. کاربران نیاز به پرداخت هزینه به ازای استفاده از خدمات یا منابع دارند. ابر عمومی توسط برخی از شرکت ها مانند گوگل ، آمازون و مایکروسافت در دسترس است.
- ابرهای خصوصی: این نوع ابر در داخل سازمان برای استفاده داخلی قرار دارد تا منحصراً از عملیات تجاری آن پشتیبانی کند. حفاظت بیشتر توسط ابر خصوصی نسبت به ابر عمومی کنترل می شود [۶]. بسیاری از دفاتر پزشکی، بانک‌ها و سایر سازمان‌ها از ابر خصوصی برای ذخیره داده‌های خود در این نوع ابر استفاده می‌کنند [۷].

- ابر ترکیبی: در این نوع، سرویس‌ها بین خصوصی و عمومی توزیع می‌شوند، جایی که برنامه‌های کاربردی حیاتی در شبکه سازمان ذخیره می‌شوند و سایر خدمات می‌توانند خارج از شبکه سازمان ذخیره شوند. زیرساخت ابری، سه مدل خدمات مجزا را ارائه می دهد که برخی از معیارهای مشتریان را برآورده می کند.

۳. زمان بندی

زمان‌بندی کار، بر اساس توزیع کار بر روی منابع موجود انجام می شود. هدف اصلی الگوریتم های زمان بندی در محیط غیرمتمرکز گسترش وظایف مختلف بر روی سرورها برای متعادل کردن بار است، این امر استفاده از پردازنده ها را به حداکثر می رساند و زمان اجرای کار کاربر را به حداقل می رساند. هدف اصلی برنامه ریزی منابع موجود با توجه به زمان موجود برای اجرای آن است. این کار ممکن است شامل وارد کردن یک پرس و جو، فرآیندی که پرس و جو می کند، دسترسی به نرم افزار و حافظه مورد نیاز باشد. سپس مرکز داده پرس و جوهای کاربر را در مورد الزامات ارائه شده در سرویس درخواستی و توافق سرویس ها طبقه بندی می کند.

الگوریتم های زمانبندی وظایف مختلف برای افزایش عملکرد ابر و افزایش توان عملیاتی سرورها استفاده می شود. محدودیت‌های متعددی در زمان‌بندی یک وظیفه مانند هزینه، توان عملیاتی، زمان، استفاده از منابع و مدت زمان وجود دارد [۸]. هدف اصلی زمان‌بندی وظیفه، به حداقل رساندن هزینه و زمان برای ایجاد یک نتیجه بهینه است که باعث افزایش عملکرد ابر می‌شود.

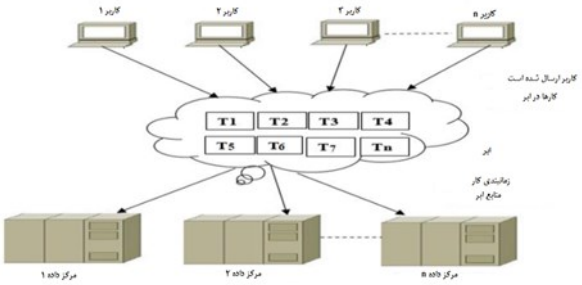
مفهوم زمان‌بندی ابری، به فناوری نگاشت مشاغل به طیفی از ماشین‌های مجازی یا اختصاص ماشین‌های مجازی برای استفاده از منابع موجود برای برآوردن نیازهای کاربر اشاره دارد. هدف از استفاده از روش‌های زمان‌بندی در رایانش ابری، افزایش توان عملیاتی سیستم و متعادل‌سازی بار، کاهش هزینه‌ها، افزایش استفاده از منابع، صرفه‌جویی در انرژی و کاهش زمان پردازش است [۹]. زمان‌بندی در دسترس بودن پردازنده و حافظه را مدیریت می‌کند. یک برنامه زمان‌بندی خوب، استفاده از منابع را افزایش می‌دهد. بسیاری از تکنیک های زمان بندی وظیفه موجود در یک سیستم محاسباتی به اندازه کافی مناسب هستند تا در سیستم ابری مورد استفاده قرار گیرند. مزیت استفاده از الگوریتم های زمان بندی وظیفه، به دست آوردن بهترین عملکرد و حداکثر توان عملیاتی سیستم است. زمان‌بندی وظیفه را می‌توان به گروه الگوریتم‌های زیر طبقه‌بندی کرد:

- الگوریتم‌های زمان‌بندی اکتشافی حالت دسته‌ای (BMHA)
- الگوریتم‌های اکتشافی حالت آنلاین
- الگوریتم اکتشافی حالت وابستگی [۱۰]

روش‌های زمان‌بندی ، کل زمان تکمیل کار را به حداقل می‌رساند، تعادل بار دستگاه را بهبود می‌بخشد و استفاده از منابع را بهبود می‌بخشد. زمان‌بندی وظایف را می‌توان به گروههای زیر از الگوریتم‌های زمان‌بندی طبقه‌بندی کرد:

- الگوریتم‌های زمان‌بندی استاتیک
- الگوریتم‌های زمان‌بندی پویا
- گردش کار
- سرویس ابری
- زمان‌بندی بلادرنگ
- زمان‌بندی اکتشافی
- تعادل بار بلادرنگ [۱۱]

کاربران، وظایف خود را به ابر ارسال می کنند و باید آنها را به پردازنده اختصاص دهند. اکنون این سوال مطرح است که وظایف پردازنده چگونه تخصیص داده می شود که دارنده ابر، کمترین زمان و حداکثر سود را کسب کند. برنامه نویسی کار مشکلی را که وظایفی به پردازنده اختصاص داده می شود که عامل دیگر را در نظر می گیرد، برطرف می کند. شکل زیر، برنامه سیستم ابری یک ماموریت را نشان می دهد [۱۲].



شکل (۱): زمان بندی وظایف[۱۲]

۴. طبقه بندی

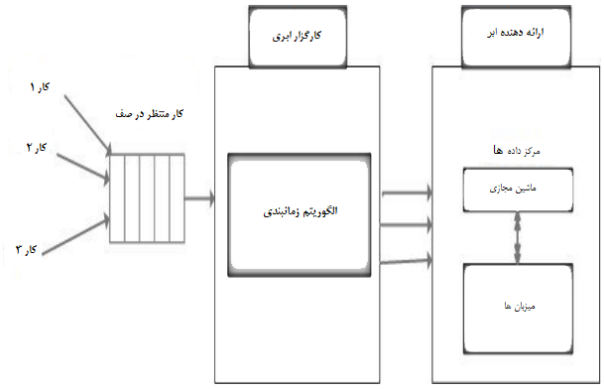
روش‌های زمان‌بندی به سه گروه اصلی طبقه‌بندی می‌شوند:

- زمان‌بندی وظایف: توزیع منابع حیاتی بین سرورها (ماشین فیزیکی) توسط زمانبندی وظایف انجام می شود. روش زمانبندی وظایف برای ساختار متمرکز و همچنین غیرمتمرکز و همچنین برای محیط همگن و ناهمگن است [۱۳].
- زمان‌بندی منابع

- زمان‌بندی گردش کار

تکنیک‌های زمان‌بندی مرسوم قادر به افزایش عملکرد نیستند و هزینه و زمان اجرا را نیز به حداکثر می‌رسانند. زمان‌بندی فرآیند برنامه‌ریزی وظایف ابری، به سه مرحله طبقه‌بندی می‌شود:

۱. فیلتر کردن منابع: ارائه‌دهنده خدمات ابری فهرستی از منابع موجود در یک شبکه خاص را کشف می‌کند و همچنین وضعیت کار آنها را جمع‌آوری و بررسی می‌کند. .
۲. انتخاب منابع: مهمترین مرحله در زمانبندی وظایف است که به عنوان مرحله تصمیم گیری نیز شناخته می شود. منابع مورد نیاز بر اساس پارامتر خاصی و با توجه به نیاز اجرای کار انتخاب می شوند.
۳. اجرای وظیفه: پس از انتخاب منبع مورد نیاز، کار برای اجرا به منبع ارائه می شود [۱۴].



شکل (۲): فرآیند زمانبندی کار [۱۴]

۵. پیشینه پژوهش

S. Ravichandran و DE Naganathan سیستم جدیدی را پیشنهاد کرده بودند که این سیستم، زمانی که وظیفه جدیدی وارد می شود، کار می کند و در صف انتظار فرستاده می شود و زمانبندی وظیفه به راحتی هر کار را سفارش می دهد و تخصیص منابع برای اجرای آنها انجام می شود. الگوریتم ژنتیک برای این کار انتخاب شده است که تمام وظایف به صف ارسال می شود، زمانبندی وظیفه را از صف منابع اختصاص می دهد و وظیفه را اجرا می کند. هدف اصلی این سیستم به حداقل رساندن زمان اجرای کار و بهینه سازی استفاده از منابع است [۱۶].

V. V. Kumar و S. Palaniswami بر افزایش کارایی تکنیک‌های زمان‌بندی کار برای سرویس رایانش ابری تمرکز کردند. آنها همچنین الگوریتمی را در نظر گرفته اند که با دادن اولویت بالا به کارهای مختلف برای تکمیل آن و اولویت کمتر برای مسائل پایان وظیفه در زمان واقعی، از زمان به طور بهینه استفاده می کند [۱۷].

در سال ۲۰۱۶، ژیفنگ ژونگ و همکاران، یک رویکرد بهینه‌سازی ازدحام ذرات حریص (G&PSO) را برای حل زمان‌بندی کار معرفی کرد. یافته‌ها نشان می‌دهد که الگوریتم پیشنهادی، عملکرد هر ماشین مجازی مانند توانایی‌های

جستجوی محلی و جهانی، نرخ هم‌گرایی سریع‌تر و حجم کاری متعادل‌تر را بهبود می‌بخشد. بنابراین، بهینه سازی ذرات حریص، از PSO کلاسیک از نظر بهره وری بیشتر از منابع بهتر عمل می کند [۱۸].

در سال ۲۰۱۷، آشیش گوپتا و ریتو گرگ، یک روش فراابتکاری از الگوریتم بهینه‌سازی کلونی مورچه‌ها برای حل زمان‌بندی کار در سیستم ابری پیشنهاد کرد که عمدتاً بر روی دو هدف تمرکز می‌کند، یعنی کاهش زمان انجام محاسبات و بهینه سازی تعادل بار. نتایج مطالعه نشان می‌دهد که الگوریتم بهینه‌سازی کلنی مورچه‌های متعادل کننده بار (LB-ACO) نتایج بهتری نسبت به الگوریتم NSGA-II در بارگذاری و بار متعادل ایجاد می‌کند [۱۹]

یک الگوریتم زمان‌بندی وظیفه جدید توسط Z. Zheng طراحی شد که مبتنی بر الگوریتم ژنتیک است که به عنوان الگوریتم ژنتیک موازی نامیده می‌شود. هدف اصلی این الگوریتم بهینه سازی ریاضی مسئله زمان بندی ابری است. سیاد بن آلا و هیچم بن آلا در یک تکنیک جدید برنامه ریزی کار پویا مبتنی بر الگوریتم ژنتیک بهبود یافته را توضیح داده اند. کار الگوریتم پیشنهادی، کاهش زمان اجرا، بهبود موثر توان عملیاتی و مقیاس‌پذیری ابر در زمان‌بندی کار متمرکز است [۲۰].

در سال ۲۰۱۸، یک الگوریتم ترکیبی بهینه‌سازی گرگ خاکستری ژنتیکی (GGWO) برای حل و بهبود مسائل زمان‌بندی کار پیشنهاد شد. نتایج به‌دست‌آمده، نشان می‌دهد که روش پیشنهادی زمان محاسبات، هزینه انتقال و استفاده از بار را در مقایسه با الگوریتم‌های QWO و GA کاهش می‌دهد [۲۱].

در سال ۲۰۱۹، فانگ ییقو و همکارانش کروموزوم‌های کدگذاری‌شده دودویی برای زمان‌بندی منابع به عنوان یک استراتژی برای بهینه‌سازی زودرس خطر جهش متقاطع در الگوریتم‌های ژنتیک تطبیقی (AGA) را پیشنهاد کردند. در این پژوهش، الگوریتم ژنتیک بهبود یافته (AGA) با الگوریتم ژنتیک استاندارد (SGA) مقایسه شده است. یافته ها نشان می دهد که رویکرد بهبود یافته، دارای یک الگوریتم برنامه ریزی موثر است [۲۲].

در سال ۲۰۱۹، پینگ ژو و همکاران، یک الگوریتم زمان‌بندی با آگاهی از قدرت و زمان واقعی (PRTS) برای کاهش هزینه‌های گردش کار و کاهش مصرف انرژی پیشنهاد داد. بخش هایی از الگوریتم پیشنهادی به شرح ذیل است:

- زمان‌بندی مقرون‌به‌صرفه‌ترین ماشین‌های مجازی بر اساس مسیر بحرانی بدون از دست دادن زمان،
- ردیابی ضعف در پویایی سیستم
- بازیابی آنها برای اجرای تکنیک صرفه‌جویی در انرژی DVFS [۲۳].

در سال ۲۰۱۹، الگوریتم بهینه‌سازی ازدحام ذرات گسسته (IDPSO) بهبود داده شد و از یک رویکرد بهینه‌سازی وزن اینرسی پویا مبتنی بر استراتژی سینوسی استفاده شد تا ذرات را با فازهای مختلف در جستجوی راه‌حل بهینه جهانی تطبیق دهند. نتایج، نشان می‌دهد که رویکرد IDPSO بر اساس زمان تکمیل و هم‌گرایی از الگوریتم DPSO و FCFS بهتر عمل می‌کند [۲۳].

در سال ۲۰۲۰، Sanj M S و همکاران، یک تکنیک Round Robin (ERR) بهبود یافته را برای افزایش کارایی بدون تأثیر بر عملکرد کلاسیک RR پیشنهاد کرد. نتایج نشان می‌دهد که کل زمان انتظار برای وظایف در تعداد مشخصی از کلودولت‌ها در ERR در شرایط مشابه در مقایسه با RR کلاسیک به حداقل می‌رسد [۲۵].

در سال ۲۰۲۰، یک تخصیص کارآمد منبع با امتیاز (ERAS) برای زمان‌بندی وظایف در محیط‌های ابری پیشنهاد شد که با پیشنهاد انواع مختلف تاخیر و EFT برای تنظیم پردازنده برای وظایف زمان‌بندی، ماشین‌های مجازی (VM) در دسترس بودن عملیاتی موقت را در نظر می‌گیرد. نتایج به‌دست‌آمده نشان می‌دهد که قابلیت اطمینان بهبود یافته الگوریتم ERAS در مقایسه با رویکردهای فعلی که فقط EFT را برای تخصیص در نظر می‌گیرند، کارایی بهتری را ارائه می‌دهد [۲۶].

در سال ۲۰۲۰، ژونگ زونگ و همکاران، ترکیبی از روش برنامه‌ریزی ماموریت همجوشی پویا، رویکرد ژنتیکی و سیستم کلونی مورچه‌ها را پیشنهاد کرد. این امر، مصرف انرژی داده ها و امکانات ذخیره سازی رایانش ابری را محدود می کند. نتایج شبیه سازی نشان می‌دهد که رویکرد پیشنهادی در برنامه‌نویسی وظیفه، زمان و انرژی دستگاه رایانش ابری را به میزان قابل توجهی به حداقل می‌رساند [۲۷].

۶. الگوریتم های تعریف وظایف

۱. الگوریتم ژنتیک: در این الگوریتم، یک جمعیت اولیه وجود دارند که وظایف را به صورت تکراری انجام می دهند و به همین دلیل، تعداد زیادی راه حل در حال رفع شدن هستند، چنین راه حل هایی به عنوان جمعیت نامیده می شوند. هر تکرار خاص در آن جمعیت به عنوان عملکرد کروموزوم نامیده می شود. ده کروموزوم از آن جمعیت انتخاب می شود [۲۸]. از این جمعیت اولیه، ده کروموزوم به صورت غیر سیستماتیک انتخاب می شوند و توابع زیر اجرا می شود:

- تابع تناسب: هدف اصلی این تابع، ارزیابی کیفیت هر فرد در جمعیت است؛ در حالی که رویکرد اصلی مربوط به بهینه سازی
- انتخاب: در این فرآیند از یک عملگر به نام انتخاب متناسب استفاده می شود که برای ارزیابی احتمال و تناسب بین دو الگوریتم استفاده می شود و مشخص می کند که احتمال انتخاب شده یا گروه های بعدی متناسب با تناسب اندام هر فرد است [۲۸].
- متقاطع: هدف از این فرآیند، انتخاب بهترین جفت افراد است و این کار، بدون استفاده از اپراتور معروف به اپراتور متقاطع تک نقطه ای انجام نمی شود. مزیت عبور این است که می توان بخش هایی از هر دو طرف را مبادله کرد.
- جهش: افراد جدید به روش آسانی برای این منظور ایجاد نمی شوند. مقداری از جایگاه ژنی با مقادیر جایگاه ژنی دیگر جایگزین می شود و در سری کدگذاری انجام می شود [۲۹].

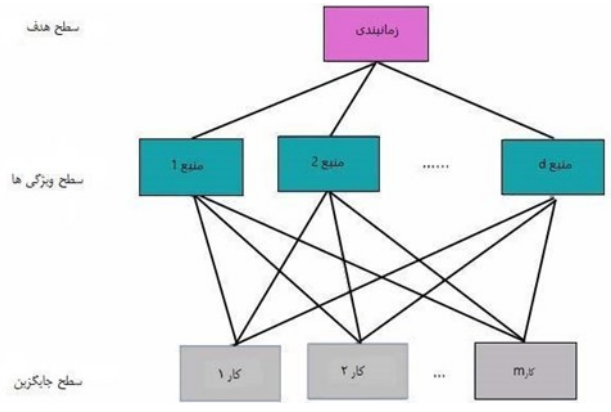
۲. الگوریتم حریصانه: برای حل مسائل با اتخاذ تصمیماتی که در آن موقعیت خاص در نظر گرفته می شود، استفاده می شود. مسائل بهینه سازی را می توان به راحتی با کمک این الگوریتم حل کرد. اگرچه برخی از مسائل با راه حل های کارآمد به اندازه کافی آسان به نظر نمی رسند، اما با کمک الگوریتم حریصانه با بهترین راه حل ها حل می شوند [۳۰]. برخی از شاخه های الگوریتم حریص عبارتند از:

- الگوریتم های حریص خالص
- الگوریتم های حریص متعامد
- الگوریتم های حریص آرام

هدف اصلی این الگوریتم، غلبه بر چنین مشکلاتی در هر مرحله و گرفتن بهترین تصمیمات بود. هدف اصلی آن، دستیابی به بهترین راه حل و ادامه کار با آن برنامه است [۳۰]. با توجه به این بهینه سازی، مسئله بزرگ به مسائل با اندازه کوچک تقسیم می شود و این کار به شناسایی راه حل ها در زمان کمتر کمک میکند [۳۱].

برخی از استانداردهای الگوریتم حریص به شرح زیر است:
- درخت پوشای حداقل کروسکال (MST): در این MST ، انتخاب یال ها به صورت جداگانه ایجاد می شود. انتخاب حریص، همیشه انتخاب لبه سبک ترین وزن است؛ زیرا چرخه ای در MST ایجاد نمی کند.
- حداقل درخت پوشای MST: مدیریت دو مجموعه انجام می شود: مجموعه ای از رئوس که قبلاً در MST جمع شده اند و مجموعه ای از رئوس که هنوز اضافه نشده اند و لبه هایی انتخاب می شوند که وزن کمتری دارند.
- کوتاه ترین مسیر: Dijkstra بسیار شبیه به الگوریتم پریم است. در این الگوریتم، کوتاه ترین درخت مسیر توسط هر لبه ساخته می شود. دو مجموعه مدیریت می شود: مجموعه ای از رئوس که قبلاً در MST جمع می شوند و مجموعه ای از رئوس که هنوز اضافه نشده اند. انتخاب حریصانه انتخاب کوچکترین مسیر وزنی است.
- تکنیک فشرده سازی بدون تلفات: کدگذاری هافمن به عنوان پایه این الگوریتم در نظر گرفته شده است. کدهای بیت با طول متغیر را به کاراکترهای مختلف اختصاص می دهد. انتخاب حریص این است که کمترین طول بیت کد را به متداول ترین کاراکتر اختصاص دهیم [۲۹].

۳. الگوریتم زمان‌بندی براساس اولویت در محاسبات ابری: یک رویکرد نوآورانه برای مقابله با کار برنامه‌نویسی توسط شمس‌الله قنبری و محمد عثمان است که با استفاده از اندازه‌گیری‌های ریاضی ارائه شده است [۳۲]. الگوریتم زمان بندی مشاغل مبتنی بر اولویت "PJSC" نامیده می شود. این، به عنوان مدل تصمیم گیری استانداردهای ضربی متمرکز است. در این الگوریتم، سه سطح اولویت وجود دارد که عبارتند از سطح برنامه نویسی، سطح منبع و سطح کار که در شکل ۵ نشان داده شده است.



شکل (۳): سه سطح از برنامه ریزی مبتنی بر اولویت[۳۲]

۴. بهینه‌سازی ازدحام ذرات (PSO): این الگوریتم، مبتنی بر جمعیت فراابتکاری است که با شیوه‌های اجتماعی پرورش ماهی و گله پرندگان به وجد آمده است. الگوریتم، شامل مجموعه ای از ذرات است و هر ذره راه حلی را برای مسئله در

فضای جستجوی معین به تصویر می کشد که سپس برای نزدیک شدن به راه حل های راحت استفاده می شود [۳۲]. این الگوریتم، با مجموعه ای از ذرات تصادفی اولیه و سپس یافتن بهترین راه حل در فضای مسئله آغاز می شود. در PSO ما از تکرار برای یافتن موقعیت هر ذره استفاده می کنیم که به عنوان بهترین سرب شخصی بدست آمده توسط ذره و بهترین موقعیت عمومی شناخته می شود.

تمام وظایف در یک محیط ابری در صف قرار می گیرند، سپس از الگوریتم بهینه سازی برای محاسبه حداقل مقادیر زمان انتظار برای همه مشاغل استفاده می شود. این حداقل‌ها برای ترتیب صحیح کارها استفاده می‌شوند که در عوض زمان کلی انتظار را به حداقل می‌رسانند [۳۳]. پس از بدست آوردن بهترین ترتیب بهینه کار، الگوریتم تولید صف برای یافتن آستانه اعمال می شود و سپس یک کار به این صف ارسال می شود. سپس برنامه‌ریز یک کار را برای یک منبع (سرور) مناسب زمان‌بندی می‌کند.

۵. الگوریتم First Come First Serve : از اساسی ترین و ساده ترین تکنیک هایی است که از زمان رسیدن کار برای برنامه ریزی کار در محیط ابری استفاده می کند. این وظیفه، با توجه به اینکه کدام وظیفه اول در صف رسیده است برنامه ریزی و اجرا می شود و کاملاً به زمان رسیدن بستگی دارد و هیچ پارامتر دیگری را در نظر نمی گیرد. وظایف، با انتخاب ترتیب صحیح کارها برنامه ریزی می شوند. وظیفه یا درخواست کاربر که ابتدا به مرکز داده می‌رسد، ابتدا برای اجرا به ماشین مجازی اختصاص داده می‌شود. کنترل‌کننده مرکز داده ماشین مجازی رایگان را بررسی می‌کند و سپس وظیفه‌ای را به آن ماشین مجازی اختصاص می‌دهد و سپس آن وظیفه را از صف حذف می‌کند. اگر ۳ وظیفه در محیط ابری با سه ماشین مجازی وارد شود، زمان‌بند FCFS سه وظیفه را به موازات ماشین مجازی برنامه‌ریزی می‌کند و یک وظیفه را تا زمانی که یک ماشین مجازی برای اولین برنامه رایگان شود، زمان‌بندی می‌کند.

۷. مقایسه الگوریتم های زمانبندی وظایف

برنامه ریزی وظایف، یکی از بزرگترین چالش ها در رایانش ابری است. انگیزه اصلی زمان‌بندی کار، توزیع وظایف ورودی از کاربران به ماشین‌های مجازی موجود با در نظر گرفتن پارامترهای مختلف تعادل بار ، زمان اجرا، تعادل بار، کیفیت خدمات، عملکرد، زمان پاسخ و تخصیص عادلانه منابع است که می‌تواند اعدام شود. برخی از الگوریتم ها فقط تعادل بار را در نظر می گیرند در حالی که برخی زمان پاسخ را در نظر می گیرند. از آنجایی که اکثر الگوریتم ها با یک یا دو پارامتر کار می کنند، به همین دلیل نمی توان نتیجه خوبی را به طور موثر بدست آورد. نتایج بهتری را می توان با جفت کردن معیارهای زمان بندی بیشتر برای ایجاد یک الگوریتم کارآمد به عنوان یک نقطه بهبود، تولید کرد، اما این کار کمی پیچیده است.

مسائل زمانبندی چالش برانگیز زیادی در زیرساخت های رایانش ابری مانند زمان محاسبات، تعادل بار، استفاده از منابع، هزینه و کیفیت سرویس وجود دارد. با توجه به آزمایشات ، نتایج خوبی برای بسیاری از الگوریتم ها در مقایسه با الگوریتم های دیگر در شرایط مشابه وجود دارد. به عنوان مثال، نویسنده در مرجع [۱۸]، یک الگوریتم (G&PSO) پیشنهاد کرد، نتایج به‌دست‌آمده نشان می‌دهد که زمان تکمیل کار کلی را در مقایسه با الگوریتم بهینه‌سازی ازدحام عملی کاهش می‌دهد؛ در حالی که نویسنده در مرجع [۱۸] الگوریتم (LB-ACO) را برای به حداقل رساندن فاصله ساخت و توزیع بار در تعادل پیشنهاد کرد، نتایج الگوریتم پیشنهادی در مقایسه با الگوریتم NSGA-II بهتر است.

۸. نتیجه گیری

الگوریتم زمان‌بندی کارآمد، می‌تواند خدمات مطلوب‌تری را به کاربران ارائه دهد و عملکرد ارائه‌شده توسط محیط ابری را افزایش دهد. هدف اصلی زمان‌بندی وظایف در محیط ابری، کاهش زمان اجرای وظایف و به حداکثر رساندن استفاده از منابع است. مشکل اصلی در زمان‌بندی کار، متعادل‌سازی بار، زمان پاسخگویی، استفاده از منابع و ذخیره‌سازی حافظه است. الگوریتم زمان بندی کارآمد را می توان با ترکیب پارامترهای مختلف با الگوریتم های موجود به دست آورد که عملکرد کلی آنها را در محیط ابری بهبود می بخشد. رایانش ابری، منبع عظیمی بر اساس نیاز مشتریان، قدرت محاسباتی، ذخیره سازی، نرم افزار و تعدادی امکانات دیگر ارائه می دهد. هدف کلیدی وظایف در تنظیمات ابری، توزیع منابع موجود برای کار در یک ساختار مناسب است. هدف اصلی برنامه نویسی، به حداکثر رساندن استفاده از منابع و کاهش زمان مصرف است. در بیشتر الگوریتم های موجود، تحلیل تطبیقی بر اساس ابعاد، فرآیندها، شرایط، نتایج و ابزارهای مختلف مورد بررسی قرار می گیرد.

۱۲. مراجع

۱. B. Kruekaew and W. Kimpan, "Multi-Objective Task Scheduling Optimization for Load Balancing in Cloud Computing Environment Using Hybrid Artificial Bee Colony Algorithm With Reinforcement Learning," in IEEE Access, vol. ۱۰, pp. ۱۷۸۰۳-۱۷۸۱۸, ۲۰۲۲, doi: ۱۰,۱۱۰۹/ACCESS.۲۰۲۲,۳۱٤۹۹۵۵.

۲. S. Nabi, M. Ibrahim and J. M. Jimenez, "DRALBA: Dynamic and Resource Aware Load Balanced Scheduling Approach for Cloud Computing," in *IEEE Access*, vol. ۹, pp. ٦۱۲۸۳-٦۱۲۹۷, ۲۰۲۱, doi: ۱۰,۱۱۰۹/ACCESS.۲۰۲۱,۳۰۷٤۱٤٥.

۳. R. Buyya, R. Ranjan, and R. N. Calheiros, “Modeling and simulation of scalable cloud computing environments and the cloudsim toolkit: Challenges and opportunities,” ۲۰۰۹, doi: ۱۰,۱۱۰۹/HPCSIM.۲۰۰۹,٥۱۹۲٦۸٥.

٤. A. Rashid and A. Chaturvedi, “Cloud Computing Characteristics and Services A Brief Review,” *Int. J. Comput. Sci. Eng.*, ۲۰۱۹, doi: ۱۰,۲٦٤۳۸/ijcse/v۷i۲,٤۲۱٤۲٦.

٥. C. Thirumalaiselvan and V. Venkatachalam, “A strategic performance of virtual task scheduling in multi cloud environment,” *Cluster Comput.*, ۲۰۱۹, doi: ۱۰,۱۰۰۷/s۱۰٥۸٦-۰۱۷-۱۲٦۸-۷.

٦. A. Abdulla, A. Abdulraheem, A. Salih, and M. Sadeeq, “(PDF) Internet of Things and Smart Home Security,” *ResearchGate*, ۲۰۲۰. .

۷. Z. S. Ageed, R. K. Ibrahim, and M. A. M. Sadeeq, “Unified Ontology Implementation of Cloud Computing for Distributed Systems,” *Curr. J. Appl. Sci. Technol.*, ۲۰۲۰, doi: ۱۰,۹۷۳٤/cjast/۲۰۲۰/v۳۹i۳٤۳۱۰۳۹.

۸. S. Parsa and R. Entezari-Maleki, “RASA: A New Task Scheduling Algorithm in Grid Environment,” *World Appl. Sci. J.*, ۲۰۰۹.

۹. D. A. Hasan, B. K. Hussan, S. R. M. Zeebaree, D. M. Ahmed, O. S. Kareem, and M. A. M. Sadeeq, “The Impact of Test Case Generation Methods on the Software Performance: A Review,” *Int. J. Sci. Bus.*, ۲۰۲۱.

١٠. I. J. Of, “A survey of various load balancing and task scheduling techniques in cloud computing,” *Int. J. Sci. Technol. Res.*, ٢٠١٦.
١١. Y. Yu and Y. Su, “Cloud Task Scheduling Algorithm Based on Three Queues and Dynamic Priority,” ٢٠١٩, doi: ١٠,١١٠٩/ICPICS٤٧٧٣١,٢٠١٩,٨٩٤٢٥٨٨.
١٢. K. Dubey, M. Kumar, and S. C. Sharma, “Modified HEFT Algorithm for Task Scheduling in Cloud Environment,” ٢٠١٨, doi: ١٠,١٠١٦/j.procs.٢٠١٧,١٢,٠٩٣.
١٣. A. G. Delavar, “RSDC (Reliable Scheduling Distributed in Cloud Computing),” *Int. J. Comput. Sci. Eng. Appl.*, ٢٠١٢, doi: ١٠,٥١٢١/ijcsea.٢٠١٢,٢٣٠١.
١٤. . P. S., “A SURVEY OF VARIOUS SCHEDULING ALGORITHM IN CLOUD COMPUTING ENVIRONMENT,” *Int. J. Res. Eng. Technol.*, ٢٠١٣, doi: ١٠,١٥٦٢٣/ijret.٢٠١٣,٠٢٠٢٠٠٨.
١٥. Y. Chawla and M. Bhonsle, “A Study on Scheduling Methods in Cloud Computing,” *Int. J. Emerg. Trends Technol. Comput. Sci.*, ٢٠١٢.
١٦. T. Goyal and A. Agrawal, “Host Scheduling Algorithm Using Genetic Algorithm in Cloud Computing Environment,” *Int. J. Res. Eng. Technol.*, ٢٠١٣.
١٧. K. Dasgupta, B. Mandal, P. Dutta, J. K. Mandal, and S. Dam, “A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing,” *Procedia Technol.*, ٢٠١٣, doi: ١٠,١٠١٦/j.protcy.٢٠١٣,١٢,٣٦٩.
١٨. Z. Zhong, K. Chen, X. Zhai, and S. Zhou, “Virtual machine-based task scheduling algorithm in a cloud computing environment,” *Tsinghua Sci. Technol.*, ٢٠١٦, doi: ١٠,١١٠٩/TST.٢٠١٦,٧٧٨٧٠٠٨.
١٩. A. Gupta and R. Garg, “Load Balancing Based Task Scheduling with ACO in Cloud Computing,” ٢٠١٧, doi: ١٠,١١٠٩/COMAPP.٢٠١٧,٨٠٧٩٧٨١.
٢٠. H. Ben Alla, S. Ben Alla, A. Ezzati, and A. Mouhsen, “A novel architecture with dynamic queues based on fuzzy logic and particle swarm optimization algorithm for task scheduling in cloud computing,” ٢٠١٧, doi: ١٠,١٠٠٧/٩٧٨-٩٨١-١٠-١٦٢٧-١_١٦.
٢١. N. Gobalakrishnan and C. Arun, “A new multi-objective optimal programming model for task scheduling using genetic gray Wolf optimization in cloud computing,” *Comput. J.*, ٢٠١٨, doi: ١٠,١٠٩٣/comjnl/bxy٠٠٩.
٢٢. F. Yiqiu and L. Xiaosheng, “Task Scheduling Strategy for Cloud Computing Based on the Improvement of Ant Colony Algorithm,” ٢٠١٨, doi: ١٠,١١٠٩/ICOMSSC٤٥٠٢٦,٢٠١٨,٨٩٤١٨٢٨.
٢٣. P. Zhu, J. Chen, and Y. G. Fu, “A power-aware scheduling algorithm for real-time workflow applications in clouds,” ٢٠١٩, doi: ١٠,١١٠٩/EITCE٤٧٢٦٣,٢٠١٩,٩٠٩٤٨٥٤.
٢٤. S. Liu and Y. Yin, “Task scheduling in Cloud Computing Based on Improved Discrete Particle Swarm Optimization,” ٢٠١٩, doi: ١٠,١١٠٩/ICISCAE٤٨٤٤٠,٢٠١٩,٢٢١٧٠٣.
٢٥. M. S. Sanaj and P. M. Joe Prathap, “An Enhanced Round Robin (ERR) algorithm for Effective and Efficient Task Scheduling in cloud environment,” ٢٠٢٠, doi: ١٠,١١٠٩/ACCTHPA٤٩٢٧١,٢٠٢٠,٩٢١٣١٩٨.
٢٦. V. A. Lepakshi and C. S. R. Prashanth, “Efficient Resource Allocation with Score for Reliable Task Scheduling in Cloud Computing Systems,” ٢٠٢٠, doi: ١٠,١١٠٩/ICIMIA٤٨٤٣٠,٢٠٢٠,٩٠٧٤٩١٤.
٢٧. Z. Zong, “An improvement of task scheduling algorithms for green cloud

computing,” ۲۰۲۰, doi: ۱۰,۱۱۰۹/ICCSE٤۹۸۷٤,۲۰۲۰,۹۲۰۱۷۸٥.

۲۸. S. A. and F. A., “Genetic-Based Task Scheduling Algorithm in Cloud Computing Environment,” *Int. J. Adv. Comput. Sci. Appl.*, ۲۰۱٦, doi: ۱۰,۱٤٥٦۹/ijacsa.۲۰۱٦,۰۷۰٤۷۱.

۲۹. A. Majumdar, “Greedy Algorithms,” in *Compressed Sensing for Engineers*, ۲۰۱۹.

۳۰. N. Krivsha, V. Krivsha, Z. Beslaneev, and S. Butenkov, “Greedy Algorithms for Granular Computing Problems in Spatial Granulation Technique,” ۲۰۱۷, doi: ۱۰,۱۰۱٦/j.procs.۲۰۱۷,۰۱,۱۱۲.

۳۱. Z. He, “Research on improved greedy algorithm for train rescheduling,” ۲۰۱۱, doi: ۱۰,۱۱۰۹/CIS.۲۰۱۱,۲٦٥.

۳۲. S. Ghanbari and M. Othman, “A priority based job scheduling algorithm in cloud computing,” ۲۰۱۲, doi: ۱۰,۱۰۱٦/j.proeng.۲۰۱۲,۱۰,۰۸٦.

۳۳. A. Verma and S. Kaushal, “Bi-Criteria Priority based Particle Swarm Optimization workflow scheduling algorithm for cloud,” ۲۰۱٤, doi: ۱۰,۱۱۰۹/RAECS.۲۰۱٤,٦۷۹۹٦۱٤.