تاریخ دریافت مقاله: ۹۵/۱۱/۱۵
تاریخ پذیرش مقاله: ۹۶/۰۳/۲۲

زمان‌بندی کارهای اولویت‌دار دارای مهلت زمانی در رایانش ابری به کمک الگوریتم بهینه‌سازی گرگ خاکستری

الهام غریبی
گروه مهندسی کامپیوتر، واحد تبریز، دانشگاه آزاد اسلامی، تبریز، ایران
پست الکترونیکی: el.gharibi92@gmail.com

نیما جعفری نویمی پور*
گروه مهندسی کامپیوتر، واحد تبریز، دانشگاه آزاد اسلامی، تبریز، ایران
پست الکترونیکی: jafari@iaut.ac.ir

چکیده

در محیط‌های ابری، منابع پردازشی متنوعی در اختیار کاربـــران قرار دارد و آن‌ها فقط برای منابعی که اســـتفاده نموده‌اند هزینـــه می‌پردازند. زمان‌بندی وظایف و مصرف انـــرژی از جمله مســـائل مهم در رایانش ابری می‌باشـــند که بســـیاری از محققان بر روی این مســـائل کارکرده‌اند. زمان‌بندی وظایف یک مسئله بهینه‌سازی NP-Hard است و بسیاری از الگوریتم‌های فرامکاشفه‌ای برای حل این مسائل ارائه‌شده اســـت. در این سیستم‌ها ســـازوکار زمان‌بندی شـــامل دو مرحلـــه اســـت: اولویت‌بندی کارهـــا و انتخاب ماشـــین‌مجازی. چالشـــی که به عنوان حیطـــه مطالعه در این مقاله انتخاب‌شـــده مسئله زمان‌بندی وظایف در محیط ابر اســـت. در این مقاله، یک روش زمان‌بندی با اســـتفاده ازالگوریتم گرگ خاکستری با در نظرگرفتن اولویت و مهلت زمانی هر وظیفه و تخصیص نزدیکترین ماشین مجازی به وظیفه موردنظر صورت گرفته که مصرف انرژی و زمان اجرای کل وظایف را به حداقل می‌رساند. نتایج شبیه‌سازی نشان می‌دهد که الگوریتم پیشنهادی بهتر از الگوریتم بهینه کلونـــی مورچگان عمل نموده و میـــزان مصرف انرژی را ۱۸٪ و زمان اجرای کل وظایف را تا ٤٪ بهبود می‌بخشد.

واژه‌های کلیدی: زمان‌بندی کارها، الگوریتم بهینه گرگ خاکستری، رایانش ابری

۱. مقدمه

رایانش ابری مدلی بر پایه شبکه‌های عظیم کامپیوتری مانند اینترنت اســـت که با به‌کارگیری اینترنت، الگویی تازه برای عرضه، مصرف و تحویل خدمت‌های فناوری اطلاعات و سایر منابع اشتراکی با به‌کارگیری شبکه ارائه می‌کند [۱-۳]. به واسطهٔ رایانش ابری و خدمات آن، کاربران می‌توانند به برنامه‌های کاربردی از برجایی از دنیا دسترسی داشته باشند [٤-٦]. در چنین الگویی دسترسی کاربران به خدمات بر اســـاس نیازهای آن‌ها و بدون توجـــه به مکان و نحوه

* نویسنده مسئول

خدمت‌دهـــی صورت می‌گیـــرد[۷-۸]. الگوی ابر به‌صورت تجاری توســـط شـــرکت‌های مهم مانند گوگل، آمازون و مایکروســـافت پشتیبانی می‌شود و این پشتیبانی دلالت بر استفاده از منابع رایانشی متشکل از سخت‌افزار و نرم‌افزار دارد که به‌صورت خدمت بر روی شـــبکه عرضه می‌شود. سیســـتم‌های کنونی رایانـــش ابری حجم گســـترده‌ای از خدمات مجازی برحسب تقاضا را پیشنهاد می‌دهند[۹-۱۰]. ارائه‌دهندگان رایانش ابری خدمات ابری را در ســـه گروه دسته‌بندی کرده‌اند[۱۱]: نرم‌افزار به‌عنوان خدمت، بُن سازه به‌عنوان خدمت، زیرســـاخت به‌عنوان خدمت. زیرســـاخت به‌عنـــوان خدمت، زیرســـاخت‌های موردنیـــاز را در قالب خدمت ارائه می‌دهند. مشـــتریان به‌جای خرید سخت‌افزار، فضای مرکز داده و تجهیزات شبکه آن‌ها را به شکل خدمت خریداری می‌کنند. بُن سازه به‌عنوان خدمت، یک بُن سازه رایانشی فراهم می‌کند که از زیرســـاخت‌های ابر استفاده می‌کند. با این ســـرویس توسعه‌دهندگان می‌توانند از تمام سیســـتم‌ها و محیط‌هایی که برای چرخـــه حیات نرم‌افزار لازم اســـت اســـتفاده کنند. از آن جمله می‌توان به توسعه، آزمـــودن، پیاده‌ســـازی و میزبانی برنامه‌هـــای تحت وب اشاره کرد. نرم‌افزار به‌عنوان خدمت، یک نرم‌افزار از طریق اینترنت ارائه می‌کند و بدین‌وســـیله نیاز به نصب نرم‌افزار روی سیستم‌های مشـــتریان از بین می‌رود و نگهداری و پشتیبانی از آن‌ها آسان‌تر می‌شود.

مسئله زمان‌بندی[۱۲-۱۳] چالش مهم دیگری در رایانش ابری است. یک الگوریتم زمان‌بند کار بایستی اولویت کارها را در نظـــر بگیـــرد زیرا برخی کارها بایـــد زودتر از دیگر کارها پردازش شوند. با افزایش تعداد کاربران ابر، وظایفی که باید زمان‌بندی شوند نیز افزایش می‌یابد. زمان‌بندی در ابر سازوکاری است که وظایف کاربران را به منابع مناسب برای اجرا تخصیص می‌دهد و به‌طور مســـتقیم بر عملکرد ابر تأثیر می‌گذارد[۱۴].

هدف از تدوین این مقاله زمانبندی مناســـب وظایف در ابر می‌باشد. از آنجایی که منابع موجود در ابر در هر زمان در حال تغییر هستند مسئله زمان‌بندی امر مهمی است که تاثیر زیادی در عملکرد محیط رایانش ابری دارد. الگوریتم زمان‌بندی روشـــی است که به وسیله آن وظایف به منابع موجود در مراکز داده تخصیص داده می‌شـــود. به‌صورت خلاصه اهداف مقاله به‌صورت زیر است:

- کاهش مصرف انرژی فراهم کنندگان ابری با زمانبندی مناسب کارها به کمک الگوریتم بهینه‌سازی گرگ خاکستری
- کاهـــش زمان اجـــرای کل کارها در رایانـــش ابری با زمانبندی مناسب آن‌ها به کمک الگوریتم بهینه‌سازی گرگ خاکستری

در بخـــش دوم برخـــی از روش‌هـــای مهـــم در زمینه زمان‌بندی و در بخش ســـوم راه کارهای گذشته مطالعه و بررســـی می‌شود و سپس در بخش چهارم انواع مدل‌های توصیفی مورد اســـتفاده در این مقاله بیان می‌شـــود و در بخش پنجم به بیان روش پیشـــنهادی، بررسی عملکرد آن بر روی مسئله بهینه‌ســـازی زمان‌بندی کارها در رایانش ابری بـــا مهلت زمانی به کمک الگوریتم گرگ خاکســـتری بهینه، با در نظرگرفتن به حداقل رساندن مصرف انرژی و در بخش ششم به جمع‌بندی نتایج و پیشنهاد کارهای آتی پرداخته شده است.

۲. زمان‌بندی وظایف در رایانش ابری

زمان‌بنـــدی از چالش‌هـــای مهـــم در رایانـــش ابری و سیستم‌های ناهمگن می‌باشـــد و به همین دلیل مطالعات و تحقیقات زیـــادی روی آن انجام‌گرفته اســـت[۱۵-۱۸]. در محیط ابری، زمان‌بندی وظایف و تخصیص منبع، توســـط ارائه‌دهنـــدگان خدمات از طریق فنـــاوری مجازی مدیریت می‌شود. زمان‌بندی وظیفه به دلیل شفافیت و انعطاف‌پذیری سیستم ابری و نیازهای مختلف برای منابع، پیچیده‌تر نیز می‌شوند. راهبردهای زمان‌بند وظیفه بر عدالت یا بهره‌وری منابـــع تمرکز دارد که هزینه، زمان، فضـــا و افزایش توان عملیاتی و کیفیت خدمت در رایانش ابری را بهبود می‌بخشد. ارائه زمان‌بندی بهینه برای کاربران و در همان زمان، ارائه

توان عملیاتی سیستم ابر و کیفیت خدمت ازجمله اهداف زمان‌بندی وظایف در ابر می‌باشد. اهداف خاص زمان‌بندی شامل: تعادل بار، کیفیت خدمات، اصول اقتصادی، بهترین زمان اجرا و توان عملیاتی سیستم می‌باشند[۱۹].

معماری زیرساخت‌های زمان‌بندی با توجه به مقیاس‌پذیری، استقلال و عملکرد سیستم بسیار مهم است. این معماری را می‌توان به سه دسته تقسیم کرد: متمرکز، توزیع‌شده و غیرمتمرکز. در زمان‌بندی متمرکز تصمیم‌گیری‌های زمان‌بندی توسط کنترل کننده مرکزی برای تمام وظایف گرفته می‌شود. زمان‌بند تمام اطلاعات را در مورد وظایف نگهداری و وضعیت همه منابع موجود در سیستم را دنبال می‌کند. زمان‌بندی متمرکز به سادگی قابل اجرا و توسعه است ولی در محیط‌های توزیع‌شده مثل ابر ناکارآمد است. زمان‌بندی توزیع‌شده شامل یک مدیرمرکزی و چند نهاد سطح پایین‌تر است. این مدیرمرکزی مسئول اجرای کامل یک وظیفه و تخصیص وظایف منحصربه فرد به ارائه دهندگان سطح پایین است. هریک از موجودیت‌های سطح پایین‌تر زمان‌بند، مسئول نگاشت وظایف به منابع ابری است. از آنجایی که موجودیت‌ها برای توسعه به سیاست‌های زمان‌بندی مدیرمرکزی نیاز دارند، بنابراین شکست مدیرمرکزی باعث خرابی در کل سیستم می‌شود. در زمان‌بندی غیرمتمرکز نیز فرض می‌شود که هریک از موجودیت‌ها مستقل هستند و تصمیم‌گیری زمانبندی از سیاست‌های خودشان ناشی می‌شود. با این حال، تصمیم‌گیری‌های مستقل ممکن است بهینه‌سازی اهدافشان را به جای عملکرد سیستم به عنوان هدف کلی در نظر بگیرند. در این مواقع، از مدل‌ها و تکنیک‌هایی استفاده می‌کنند که کنترل واحدهای خود را داشته باشند و به طور همزمان عملکرد عمومی سیستم را نیز نگه می‌دارند[۱۹].

زمان‌بندی وظایف به دو نوع ایستا و پویا تقسیم‌بندی می‌شود. در زمان‌بندی ایستا، قبل از اجرای برنامه و زمان‌بندی آن تمام اطلاعات مانند ساختار برنامه، مدت‌زمان‌های اجرای وظایف و هزینه‌های ارتباطات مابین آن‌ها مشخص است ولی در زمان‌بندی نوع پویا این اطلاعات تا زمان اجرا مشخص نیست[۲۰].

۳. مطالعه و بررسی روش‌های موجود برای زمان‌بندی در ابر

در این قسمت نوع کارکرد برخی از کارهای مرتبط در زمینه زمان‌بندی وظایف بر روی سیستم‌های ابر و ناهمگن موردمطالعه و بررسی قرار می‌گیرد. ماتئوس و همکاران [۲۱] در مقاله‌ای یک الگوریتم زمان‌بند جدید مبتنی بر الگوریتم کلونی مورچه را ارائه نموده‌اند. هدف این زمان‌بند کمینه کردن زمان گردش مجموعه کارها هم‌زمان با کمینه کردن حداکثر زمان اتمام انجام کل کارها می‌باشد. نتایج شبیه‌سازی‌شده با داده‌های واقعی و سایر سیاست‌های زمان‌بندی ابر نشان می‌دهد که این روش ارائه‌شده اجازه پرداختن به کارهای چابک‌تر را درحالی‌که زمان تکمیل کاهش می‌یابد می‌دهد. همچنین نتایج ارزیابی نشان می‌دهد که الگوریتم کلونی مورچه بهتر از الگوریتم تصادفی می‌باشد. از مزایای این الگوریتم پیچیدگی و هزینه ارتباطی کمتر و از معایب آن، نتایج نامناسب در گراف‌های بر پایه محاسبات است. سلوارانی و همکاران[۲۲] یک الگوریتم مبتنی بر هزینه برای نگاشت کارآمد وظایف به منابع در دسترس ابر ارائه کرده‌اند. این الگوریتم زمان‌بندی هم هزینه استفاده از منبع و هم‌کارایی رایانشی را اندازه‌گیری می‌کند. الگوریتم ارائه‌شده همچنین نسبت رایانش به ارتباطات را با استفاده از گروه‌بندی وظایف کاربران بر اساس ظرفیت پردازش منابع ابر و ارسال کارهای گروه‌بندی‌شده به منابع بهبود می‌بخشد. الگوریتم زمان‌بندی تمام وظایف کاربران را بر اساس اولویت آن‌ها به سه دسته اولویت بالا، اولویت متوسط و اولویت پایین تقسیم می‌کند. از مزایای این الگوریتم بهبود هزینه پردازش و زمان پاسخ و معایب الگوریتم مبتنی بر هزینه، عدم تولید نتایج پایدار در مسائل با طیف گسترده است.

مزماز و همـــکاران[۲۳] یک الگوریتم زمان‌بندی ژنتیک ترکیبی دو هدفه برای برنامه‌های کاربردی موازی اولویت محدود در سیستم‌های توزیع‌شده ناهمگن مانند زیرساخت رایانش ابری پیشنهاد داده‌اند.از مزایای این الگوریتم کاهش زمان اتمام وظایف و هزینه ارتباطی است، در این الگوریتم تمرکز بر روی مدل‌های موازی جزیره‌ای و چند مدل شروع موازی است، از معایب روش تولید جمعیت اولیه به‌صورت تصادفی است. روش جدید که بر اساس مقیاس ولتاژ پویا مصرف انرژی را کاهش می‌دهد با اســتفاده از پیشــرفت اخیر در طراحی پردازنده این کار صورت می‌گیرد.

۴. مدل‌ها

در ایــن بخــش، توصیف مدل‌سیســتمی، انــرژی و جریان‌کاری مورداستفاده در این مقاله بیان می‌شود.

٤-١ مدل سیستم

در این تحقیق، فرض شــده که سیستم موردنظر شامل مجموعــه‌ای از پردازنده‌هــای ناهمگن اســت: $P = \{p_i\}$ که هریــک به‌صورت کامل به یک همبندی به‌طور داخلی متصل هستند[۲٤]. فرض شــده که مجموعه‌ای از گراف‌های وظیفه به‌صورت زیر اســت: $N = \{n_i\}$. هــر پردازنده p_j که عضو $p_j \in P$: P، ولتاژ مقیاس‌پذیر پویا که قادر اســت در سطوح مختلف ولتاژی و بسامد زمان‌ســنجی عمل نماید. برای هر پردازنده $p_j \in P$ مجموعه‌ای متوالی از ولتاژ ذخیره‌شــده را به‌صورت $V = \{v_s\}$ تعریف می‌نماییم و بسامد زمان‌سنجی، به‌صورت مجموعه $F = \{f_s\}$ تعریف می‌شود که در مدل سیستم کنونی فرکانس بسیار ناچیز در نظر گرفته‌شده است. زمانی که ولتاژ ذخیره در سطح v_1 عمل می‌کند، بسامد زمان‌سنجی در سطح f_1 عمل خواهد کرد. با توجه به این‌که ماشین هنوز انــرژی مصرف می‌کند و در حالت ایــده‌آل قرار دارد، چنین ماشینی در کمترین حالت مصرف ولتاژ v_{lowest} با بیشترین میزان ذخیره انرژی قرار دارد. در این مقاله، همپوشانی انتقالات بســامدی را برای مدت زمان قابل اغماض، در نظر نخواهیم گرفت[۲۵].

٤-٢ مدل جریان کاری

به‌طورکلــی، کاربردهای جریان کار موازی را می‌توان توســط یک گراف غیر حلقوی(DAG) به‌صورت شــکل ۱ نشــان داد. گراف وظیفه G = (N,E) شــامل مجموعه‌ای از رئوس N و مجموعه‌ای از لبه‌های E اســت که در اینجا N مجموعه‌ای از وظایف تفکیک‌شــده از یک کاربرد است و E مجموعه‌ای از لبه‌های موجود مابین وظایفی است که بیانگر محدودیت‌های دارای اولویت و مهم هستند.

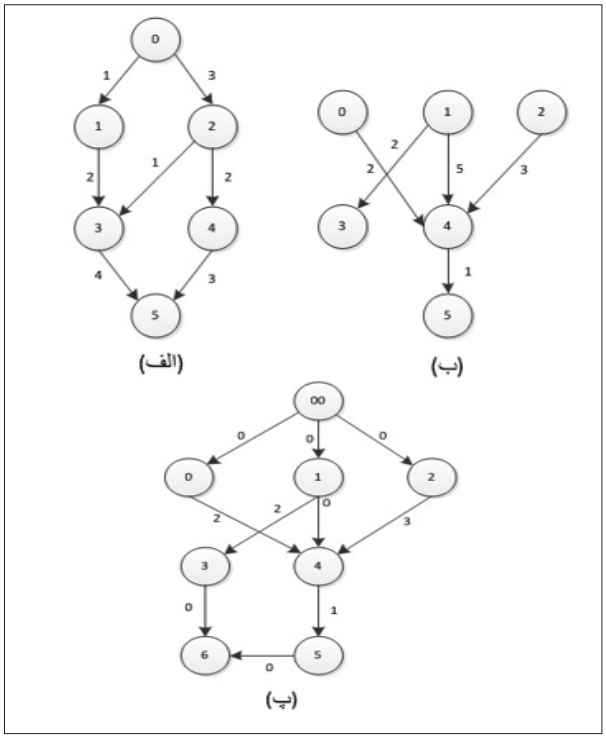
هر edge (i, j) ∈ E مابین وظایف n_i و n_j نشان می‌دهد که ارتباط درون وظیفه‌ای وجــود دارد. در کل، وظیفه n_j نمی‌تواند شــروع به کار نماید تا زمانی‌که اجرای وظیفه n_i تمام نشده باشد. یک وظیفه بدون اجداد یک وظیفه N ورودی n_{entry} نامیده می‌شــود (مانند n_0 در شکل ۱ الف) و یک وظیفه بدون جایگزین‌ها به‌عنوان وظیفه خروجی n_{exit} نامیده می‌شود (مانند n_5 در شکل ۱- الف). درصورتی‌که بیش از یک وظیفه ورودی وجود داشــته باشد(مانند n_0، n_1، n_2در شکل ۱- ب) بنابراین بیش از یک وظیفه خروجی نیز وجود خواهد داشــت (مانند n_3،n_s در شکل ۱- ب) پس لازم است که یک وظیفه ورودی مجازی تعریف شــود (مانند n_{00} در شــکل۱- پ) و یا یک وظیفه خروجی مجازی (مانند n_6 در شکل ۱- پ). در این حالت چندین وظیفه ورودی یا چندین وظیفه خروجی مرتبط خواهند بود.

این فرآیندها، گــراف DAG را که دارای یک و فقط یک وظیفه ورودی یا وظیفه خروجی است به وجود می‌آورند. وظیفه مجــازی ورودی (خروجی) یک گره با هزینه صفر است که به همه وظایف واقعی با لبه‌های هزینه صفر مرتبط است که بر زمان‌بندی وظایف تأثیری ندارد[۲٦].

زمان شــروع واقعــی (AST) برای هــر گره n_i روی پردازنــده P_k، $AST(n_i,P_k)$ به‌صــورت (۱) نشــان داده می‌شود[۲۷]:

$$AST(n_i, P_k) = \max\big(EST(n_i, P_k), Avail(P_k)\big) \quad (۱)$$

در معادله (۱)، منظور از $Avail(P_k)$ اولین زمانی اســت کــه پردازنده P_k آمــاده به اجرای وظیفه می‌باشــد. اولین



شکل ۱: نمونه یک گراف[۲۵]

زمان پایان (EFT) هر گره n_i روی پردازنده P_k، $\mathrm{EFT}(n_i, P_k)$ به‌صورت (۲) نشان داده می‌شود[۲۷]:

$$EFT(n_i, P_k) = AST(n_i, P_k) + W(n_i, P_k) \qquad (۲)$$

زمان واقعی پایان (AFT) هر گره n_i روی تمام پردازنده‌ها، $AFT(n_i, P_k)$ به‌صورت (۳) نشان داده می‌شود [۲۷]:

$$AFT(n_i, P_k) = min_{1 \le l \le m} EFT(n_i, P_l) \qquad (۳)$$

۳-٤ مدل مصرف انرژی

یک مدل انرژی[۲۸] از مدل مصرف توان مشتق شده که این ریزمدل مصرف توان تکمیل‌کننده مدار منطقی نیمه‌رسانای اکسید فلزی(CMOS) است. میزان مصرف توان یک CMOS مبتنی بر ریز پردازنده‌هایی است که به‌صورت مجموعه‌ای خازنی، مدار کوتاه و توان اندک (یا بدون توان) تعریف‌شده است. توان خازن (اتلاف توان پویا) مهم‌ترین معیار مصرف توان است. توان خازن (P_c) به‌صورت زیر تعریف می‌شود[۲۹]:

$$P_c = ACv^2 f \qquad (۴)$$

مطابق (٤)، A تعداد سوییچ‌ها در هر سیکل ساعت (زمان)، C بارگذاری کل خازن، v ذخیره ولتاژ و f فرکانس تعریف می‌شود. در (٤) به‌وضوح نشان داده‌شده که ولتاژ ذخیره‌شده یک معیار مهم و اصلی است، بنابراین کاهش در میزان آن بر مقدار مصرف توان تأثیر اساسی می‌گذارد. میزان مصرف انرژی در اجرای یک کاربرد موازی محدود به‌صورت زیر تعریف می‌شود[۲۹]:

$$E = \sum_{i=1}^{n} AC v_i^2 f . w_i^* = \sum_{i=1}^{n} \alpha\, v^2\, w_i^* \qquad (۵)$$

بر طبق معادله (۵) v_i ولتاژ ذخیره پردازنده بر اجرای وظیفه n_i است و w_i هزینه مصرف وظیفه n_i (مدت‌زمان لازم برای اجرای n_i) در پردازنده زمان‌بندی‌شده است[۳۰].

۵. روش پیشنهادی

الگوریتم گرگ خاکستری در سال ۲۰۱٤ بر مبنای زندگی گرگ‌های خاکستری ارائه‌شده است. این حیوانات به‌طور گروهی زندگی می‌کنند و رهبر گروه، آلفا مسئول تصمیم‌گیری در زمینه‌هایی مثل حمله، مکان خواب و زمان حمله است. روش شکار این‌گونه از حیوانات در این الگوریتم برای بهینه‌سازی مورد الهام قرارگرفته است. مرحله شکار گرگ‌ها در سه مرحله است:

۱) ردگیری، تعقیب و نزدیک شدن به شکار. ۲) دنبال کردن، محاصره کردن، اذیت و آزار کردن شکار تا زمانی که شکار از حرکت باز ایستد. ۳) حمله به‌طرف شکار.

زمان‌بندی وظایف، فرآیند نگاشت وظایف یک برنامه به پردازنده‌های سیستم و مرتب‌سازی ترتیب آن‌ها برای هر پردازنده است. وظایف به همراه اولویت‌های اطلاعات آن‌ها توسط گراف جهت‌دار بدون دور مدل‌سازی می‌شود. هدف و ایده اصلی از زمان‌بندی گراف جهت‌دار بدون دور کاهش زمان اجرای کل برنامه می‌باشد که زمان کلی اجرای تمام وظایف برنامه و کاهش میزان مصرف انرژی است. در روش پیشنهادی مسئله زمان‌بندی وظایف در رایانش ابری با الگوریتم بهینه‌سازی گرگ خاکستری بررسی شده است. در روش پیشنهادی مسئله زمان‌بندی وظایف در رایانش ابری با الگوریتم بهینه‌سازی گرگ خاکستری

صورت گرفته اســت. مراحل انجام الگوریتم پیشــنهادی به‌صورت زیر می‌باشد: مقداردهی تعداد وظایف (گرگ) و ماشین مجازی(طعمه). دریافت زمان ورود و زمان خدمت و مهلت‌زمانی وظایف از ورودی. مشخص نمودن موقعیت وظایف و ماشین‌های مجازی. تخصیص اولویت به وظایف. انتخاب وظایف دارای بالاترین اولویت و مهلت زمانی کمتر. محاسبه فاصله ماشین مجازی تا وظایف تخصیص ماشین مجازی براساس ویژگی وظایف. محاسبه زمان اجرای کل کارها و مصرف انرژی و تابع برازندگی. براســاس تکنیک اشــتراک زمانی، صفــی از وظایف و یک بــازه زمانی در اختیار است، به هر وظیفه اندازه آن بازه زمانی، زمان داده می‌شــود تا بر روی آن عنصر پردازشی صورت گیرد و اجرا شود. در این تکنیک وظایف به‌صورت همزمان پخش شده و یک‌جا اســتفاده می‌شــوند. بروز رسانی موقعیت وظایف براساس اولویت و مهلت زمانی هر وظیفه صورت می‌گیرد، شــرط خاتمه الگوریتم اتمام وظایف است، جواب بهینه برحسب تابع برازندگی به دست می‌آید.

۵-۱ مدل‌سازی الگوریتم گرگ خاکستری

با توجــه به توضیحات داده‌شــده در خصوص نحوه زندگی اجتماعی و شــکار گرگ خاکستری، در این بخش، مدل ریاضی نحوه رفتار اجتماعی و شکار گرگ خاکستری در جهت معرفی الگوریتم گرگ خاکســتری بیان می‌شود. به‌منظور مدل‌ســازی رفتار اجتماعی گرگ‌ها، یک جمعیت تصادفــی از راه‌حل‌ها تولید و بهترین راه‌حــل به نام آلفا (α)، دومین و ســومین راه‌حل مناســب به ترتیب به نام بتا (β) و دلتا (δ) معرفی می‌شــود. سایر راه‌حل‌ها را به‌عنوان گرگ‌هــای دســته امــگا (ω) در نظر می‌گیریــم. الگوریتم گرگ خاکســتری از ســه جواب δ، β، α جهت هدایت شکار (بهینه‌ســازی) اســتفاده می‌کند و جواب‌های ω از این سه پیروی می‌کنند. به‌منظور مدل‌ســازی ســه فاز، ابتدا نقاط اطراف طعمه را مشــخص، سپس به سمت طعمه حرکت و درنهایت به طعمه حمله می‌شود. برای مشخص کردن نقاط اطراف طعمه از روابط(۷) و (۸) استفاده می‌شود[۳۱].

$$\vec{D} = |\vec{C}.\ \overrightarrow{X_P}(t) - \vec{X}(t)| \qquad (۷)$$

$$\vec{X}(t+1) = \overrightarrow{X_P}(t) - \vec{A}.\vec{D} \qquad (۸)$$

که t نشــان‌دهنده تکرار جاری، $\vec{A}$ و $\vec{C}$ بردار ضرایب، $\overrightarrow{X_P}$ بردار موقعیت طعمه و $\vec{X}$ بردار موقعیت یک گرگ خاکستری است. بردار $\vec{A}$ و $\vec{C}$ با استفاده از روابط (۹) و (۱۰) محاسبه می‌شوند[۳۱]:

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \qquad (۹)$$

$$\vec{C} = 2\overrightarrow{r_2} \qquad (۱۰)$$

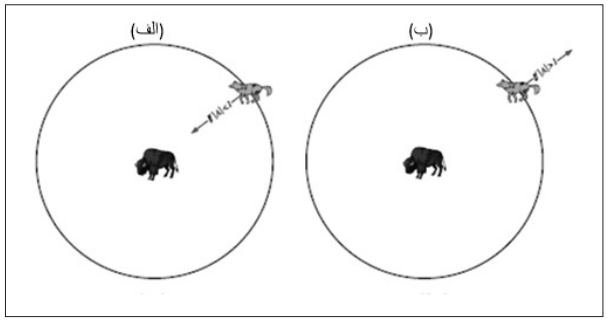
که مؤلفه‌هــای $\vec{a}$ در طول تکــرار از ۲ تا ۰ به‌صورت خطــی کاهش پیــدا می‌کننــد و r_1 و r_2 بردارهای تصادفی هســتند که در بازه $[0,1]$ می‌باشند. با توجه به این‌که در فضای جســتجوی اولیه ایده‌ای در مورد موقعیت شــکار وجود ندارد، فرض می‌شود که α (بهترین نامزد برای حل)، β و δ بهترین دانش در مورد موقعیت شکاردارند. بنابراین سه پاســخ مناسب را که تاکنون داشته است حفظ کرده و سایر عوامل از قبیل ω مجبور می‌شود تا موقعیت خودشان را با توجه به روابط (۱۱)، (۱۲) و (۱۳) جهت نزدیک شــدن به مقدار بهینه تغییر دهند[۳۱]:

$$\overrightarrow{D_\alpha} = |\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}|\ ,\quad \overrightarrow{D_\beta} = |\overrightarrow{C_2}.\ \overrightarrow{X_\beta} - \vec{X}|$$
$$\overrightarrow{D_\delta} = |\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}| \qquad (۱۱)$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.(\overrightarrow{D_\alpha})\ ,\quad \overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta})$$
$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \qquad (۱۲)$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1}+\overrightarrow{X_2}+\overrightarrow{X_3}}{3} \qquad (۱۳)$$

همان‌طــور که گفته شــد گرگ‌ها زمانی که شــکار را متوقــف کردنــد به‌طرف آن حملــه می‌کنند. بــرای این‌که چنیــن پدیده‌ای مدل شــود مقدار $\vec{a}$ کاهــش می‌یابد، بازه نوســان $\vec{A}$ به‌وســیله $\vec{a}$ کاهش می‌یابد. به‌عبارت دیگر $\vec{A}$ یک مقدار تصادفی در فاصله $[-2a,2a]$ اســت که a از ۲ به ســمت ۰ کاهش می‌یابد. زمانی که مقــدار تصادفی $\vec{A}$ در بازه $[-1,1]$ است، موقعیت بعدی یک عامل می‌تواند در هر موقعیتی بین موقعیت جاری و موقعیت شکار باشد. شکل (۲الف) نشــان می‌دهد که در حالتی که $|A|<1$ است گرگ به‌طرف شکار حمله می‌کند. در الگوریتم گرگ خاکستری،



شکل۲: حمله به طعمه در برابر جستجو برای طعمه[۳۱]

موقعیت‌ها با توجه به موقعیت آلفا، بتا و دلتا بروز رسانی می‌شود. ممکن اســت که آن‌ها برای پیداکردن شکارها از هم دور شــوند و همیشه نزدیک شدن اتفاق نمی‌افتد. برای مدل‌سازی چنین پدیده‌ای، از مقدار تصادفی بزرگ‌تر از ۱ یا کمتر از ۱- برای $\vec{A}$ اســتفاده می‌شود. شکل(۲ب) نشان می‌دهد که اگر $|A|>1$ باشد گرگ‌ها از شکار دور می‌شوند. شبه کد الگوریتم گرگ خاکســتری در شکل ۳ نشان داده شده است.

۵-۲ تابع برازندگی

در زمان‌بنــدی رایانــش ابری، اهداف کلــی به حداقل رســاندن زمان اتمام کل کارهــا و کاهش انرژی مصرفی می‌باشــد. زمان پایان واقعی آخریــن زیروظیفه در گراف جهت‌دار بدون دور یا به عبارتی گره خروجی گراف زمان اجرای کل برنامه می‌باشد. مؤلفه زمان اجرای کل برنامه از رابطه (۱٤) به دست می‌آید.

$$Makespan = max\{AFT(n_{exit})\} \qquad (۱۴)$$

الگوریتم روش پیشنهادی بر اساس مقیاس بندی ولتاژ پویا بــرای کاهش مصرف انرژی جهت بهبــود کارایی و عملکرد وظایف منعقدشــده است. مؤلفه مصرف انرژی از رابطه (۱۵) به دست می‌آید.

$$E = \sum_{i=1}^{n} \propto v^2 w_i^* \qquad (۱۵)$$

مقدار برازندگی روش پیشــنهادی بــرای بهره‌وری و بهبود مصرف انرژی و کاهش زمان اتمام کارها به‌صورت رابطه (۱٦) به دست می‌آید.

```
Initialize the grey wolf population Xi (i = 1, 2, ..., n)
Initialize a, A, and C
Calculate the fitness of each search agent
Xα=the best search agent
Xβ=the second best search agent
Xδ=the third best search agent
while (t < Max number of iterations)
    for each search agent
        Update the position of the current search agent by equation (13)
    end for
    Update a, A, and C
    Calculate the fitness of all search agents
    Update Xα, Xβ, and Xδ
    t=t+1
end while
return Xα
```

شکل ۳: شبه کد الگوریتم گرگ خاکستری[۳۱]

$$Fitness\ Function = W_1 M + W_2 E \qquad (۱۶)$$

در تابــع برازندگی فــوق W_1 و W_2 وزن اولویت‌بندی را نشان می‌دهد که در بازه صفر و یک می‌باشند.

$$W_1 + W_2 = 1 \qquad (۱۷)$$

مقادیر تابع برازندگی طبق فرمول (۱۷) نرمال‌ســازی صورت می‌گیرد، تابع نرمال‌سازی عدد حقیقی نرمال‌شده بین را برمی‌گرداند.

$$X = [a,b] \quad \Rightarrow \quad [a',b']$$

$$X' = \frac{x-a}{b-a}(b'-a') + a' \qquad (۱۸)$$

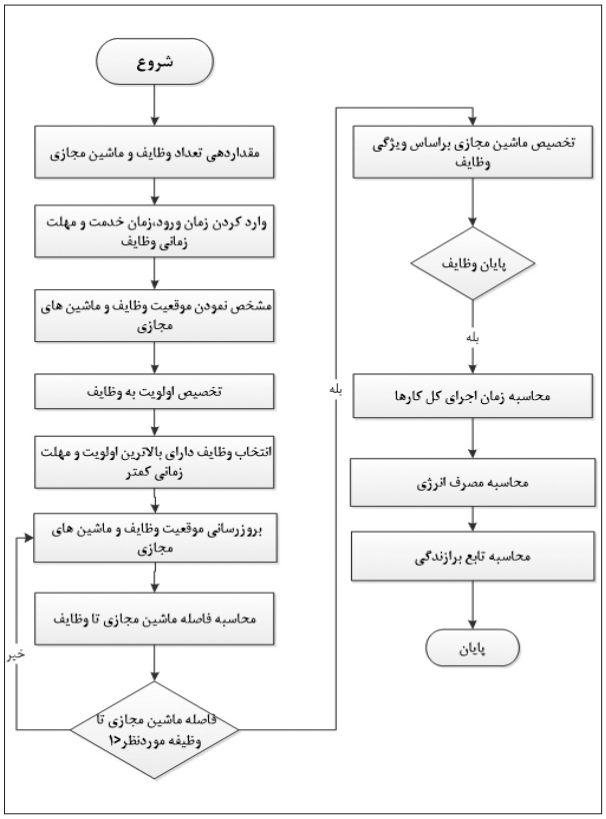
۵-۳ روندنمای روش پیشنهادی

روندنمای روش پیشــنهادی برای زمان‌بندی وظایف در رایانش ابری با اســتفاده از الگوریتم بهینه‌سازی گرگ خاکستری در شــکل ٤، مراحل اجرای روش پیشنهادی را نشان می‌دهد.

۵-٤ شبیه سازی روش پیشنهادی

در ایــن بخــش شبیه‌ســازی الگوریتم پیشــنهادی با شبیه‌ســاز کلودســیم که یکی از معروف‌تریــن ابزارهای شبیه‌ســازی محیــط رایانش ابری اســت و امکان تعریف جریــان‌کاری و میزبان‌ها را می‌دهد انجام شــده اســت و آزمایش‌های خود را با اطلاعات ورودی جدول ۱ و جدول ۲ که بــه ترتیب جدول پارامترهای شبیه‌ســازی و جدول اطلاعات ورودی هستند انجام داده‌ایم.

شبیه‌سازی و مقایسه روش پیشنهادی با الگوریتم‌های خروج به ترتیب ورود و نوبت گردشــی و کلونی موچگان از [۳۲] در محیــط کلودســیم انجام گرفته اســت. در حال



شکل ۴: فلوچارت روش پیشنهادی با استفاده از الگوریتم بهینه‌سازی گرگ خاکستری

حاضر نتایج شبیه‌ســـازی در چندین حالت بررســـی شده اســـت که به قرار زیر اســـت. روش پیشنهادی بهینه گرگ خاکســـتری براســـاس دو معیار مهم زمان کل اتمام کار و مصرف انرژی با روش های ذکرشـــده مقایسه شده است. منظور از زمان اجرای کل وظایف این اســـت که هر وظیفه از لحظه ورود به سیستم تا زمان پاسخ دهی، چه مدت در سیســـتم می‌ماند تا پاسخ خود را دریافت کند. مقدار زمان اجرای کل کارها بر حســـب ثانیه در نظر گرفته شده است. در این نمودار محور افقی نشـــان‌دهنده تعداد وظایف است و نشـــان می‌دهد که درحالت مقایسه بیشترین مقدار چقدر اســـت. تعداد وظایف بین بـــازه ۱۰۰ تا ۱۰۰۰ متغیر فرض شده اســـت. ابتدا تعداد وظایف برابر ۱۰۰ و تعداد ماشین مجازی برابر ۵۰ در نظر گرفته شده ، سپس به تعداد ۱۰ بار روش پیشنهادی اجرا شده و با نتایج [۳۲] مقایسه صورت گرفتـــه، از بین ۱۰ عددی که بـــرای زمان اجرای کل کارها حاصل شده بزرگترین مقدار محاســـبه می‌شود. با توجه

جدول ۱: اطلاعات ولتاژ پردازنده‌ها[۲۳]

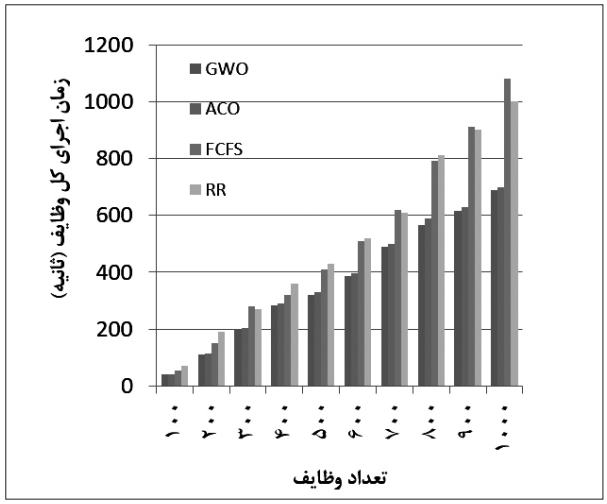
سطح	Intel Pentium M	AMD Opteron 2218	AMD Turion MT-34
	ولتاژ(ولت)	ولتاژ(ولت)	ولتاژ(ولت)
۰	۱,۵	۲,۲	۱,۷۵
۱	۱,۴	۱,۹	۱,۴
۲	۱,۳	۱,۶	۱,۲
۳	۱,۲	۱,۳	۰,۹
۴	۱,۱	۱,۰	
۵	۱,۰		

جدول ۲: پارامترهای شبیه سازی[۳۲]

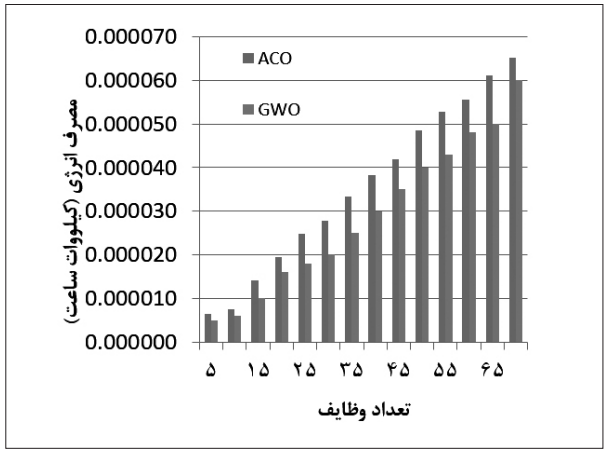
نوع موجودیت	پارامترها	مقادیر
کار	طول کار	۱۰۰۰-۲۰۰۰۰
	تعداد کارها	۱۰۰-۱۰۰۰
ماشین مجازی	تعداد ماشین مجازی	۵۰
	حافظه ماشین مجازی	۲۵۶-۲۰۴۸ KB
	پهنای باند	۵۰۰-۱۰۰۰ Mips
	نرخ پردازنده	۵۰۰-۲۰۰۰ Mips
	تعداد پردازنده	۱-۴
مرکز داده	تعداد میزبان	۲-۶

به بیشـــترین مقادیر زمان اجرای کارهای به دست آمده از این حالت‌ها می‌توان گفت روش پیشـــنهادی دارای عملکرد بهتری می‌باشـــد. در شکل ۵ نمودار مقدار زمان اجرای کل کارها در روش پیشنهادی نسبت به سایر الگوریتم‌ها بهبود یافته است.

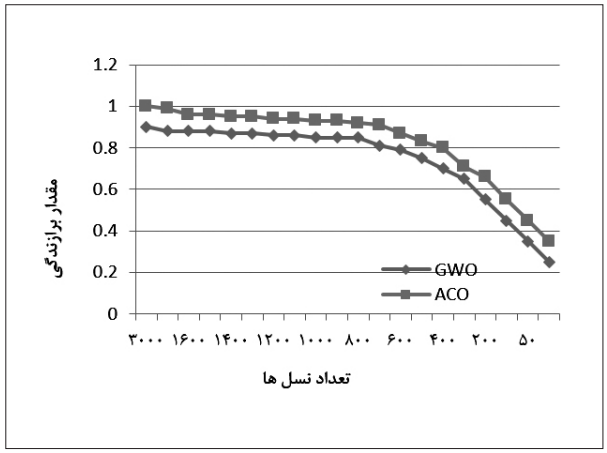
در روش پیشـــنهادی، اســـتفاده از روش مقیاس ولتاژ پویـــا و مدل کردن وظایف موجود بر روی گراف جهت‌دار بدون دور، براســـاس نتایج به‌دســـت آمده، کاهش انرژی مصرفـــی کل را در پـــی دارد. برای مقایســـه این مقدار در روش پیشـــنهادی و روش الگوریتم کلونی مورچگان[۳۳]، محور افقی برابر با تعـــداد وظایف و محور عمودی برابر مصرف انرژی وظایف در نظر گرفته می‌شـــود. در هر بار اجرا، تعداد وظایف از ۵ تا ۷۰ متغیر فرض شـــده است. در هر بازه که سیستم اجرا می‌شود، مقادیر متفاوتی به دست می‌آید که در شـــکل ٦ قابل مشاهده است. با افزایش تعداد



شکل ۵: زمان اجرای کل وظایف



شکل۶: مصرف انرژی با تعداد وظایف مختلف



شکل۷: نمودار همگرایی برای تابع برازندگی

وظایف در روش پیشـــنهادی نســـبت روش موجود مقدار مصرف انرژی به طور چشمگیری کاهش می‌یابد.

محاسبه تابع هدف و نحوه عملکرد آن برای بهینه‌سازی الگوریتم‌ها، مقیاس‌گذاری و کارایی نسبی برای تعداد نسل مختلف مورد مقایســـه قرار گرفته که در شـــکل ۷ نشـــان داده شده اســـت. این مقایسه نشان می‌دهد تابع برازندگی الگوریتم پیشنهادی همگرایی مناسبی دارد.

۶. نتیجه گیری

در این مقاله، روش جدید طراحی شده برای زمانبندی رایانش ابری با استفاده از روابط، شبه‌کد، روندنما و گراف توصیف شد. با اســـتفاده از نرم افزار شبیه‌ساز کلودسیم پیاده‌سازی انجام گرفت. نتایج به‌دست آمده از پیاده‌سازی نشـــان می‌دهد کـــه معیارهای زمان کلـــی کارها، مصرف انرژی با در نظرگرفتن مهلت زمانی کارها بهبود داده شده است. در آینده نیز می‌توان الگوریتم پیشنهادی را با توازن بار الگوریتـــم کلونی مورچه‌ها ترکیب نمود و تاثیر هزینه در کنار زمان اجرا و انرژی را ارزیابی نمود.

مراجع

[۱] م. حاجی بابا و. س. گرگین. "مقایســـه مدل های رایانشـــی توزیع شـــده و کاربردپذیری آن ها،" مجله علوم رایانشی.. شماره ۲، ۱۳۹۵.

[2] I. Saeedeh Hazratzadeh, (Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz and I. Nima Jafari Navimipour, (Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, "Colleague recommender system in the Expert Cloud using features matrix."

[3] B. Alami Milani and N. Jafari Navimipour, "A comprehensive review of the data replication techniques in the cloud environments," J. Netw. Comput. Appl., vol. 64, no. C, pp. 229–238, 2016.

[4] N. J. Navimipour, A. M. Rahmani, A. H. Navin, and M. Hosseinzadeh, "Expert Cloud: A Cloud-based framework to share the knowledge and skills of human resources," Comput. Human Behav., vol. 46, pp. 57–74, 2015.

[5] F. Sheikholeslami and N. J. Navimipour, "Service allocation in the cloud environments using multi-objective particle swarm optimization algorithm based on crowding distance," Swarm Evol. Comput., 2017.

[6] N. Jafari Navimipour, "A formal approach for the specification and verification of a Trustworthy Human Resource Discovery mechanism in the Expert Cloud," Expert Syst. Appl., vol. 42, no. 15–16, pp. 6112–6131, Sep. 2015.

[7] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, and I. Stoica, "Above the clouds: A berkeley view of cloud computing," 2009.

spired algorithm for minimizing weighted flowtime in cloud-based parameter sweep experiments," Adv. Eng. Softw., vol. 56, pp. 38–50, Feb. 2013.
[22] S. Selvarani and G. S. Sadhasivam, "Improved cost-based algorithm for task scheduling in cloud computing," in Computational intelligence and computing research (iccic), 2010 ieee international conference on, 2010, pp. 1–5.
[23] M. Mezmaz, N. Melab, Y. Kessaci, Y. C. Lee, E.-G. Talbi, A. Y. Zomaya, and D. Tuyttens, "A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems," J. Parallel Distrib. Comput., vol. 71, no. 11, pp. 1497–1508, Nov. 2011.
[24] S. U. Khan and I. Ahmad, "A cooperative game theoretical technique for joint optimization of energy consumption and response time in computational grids," IEEE Trans. Parallel Distrib. Syst., vol. 20, no. 3, pp. 346–360, 2009.
[25] Y. C. Lee and A. Y. Zomaya, "Energy conscious scheduling for distributed computing systems under different operating conditions," IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 8, pp. 1374–1381, 2011.
[26] Z. Tang, L. Qi, Z. Cheng, K. Li, S. U. Khan, and K. Li, "An Energy-Efficient Task Scheduling Algorithm in DVFS-enabled Cloud Environment," J. Grid Comput., vol. 14, no. 1, pp. 55–74, 2016.
[27] Y. Xu, K. Li, J. Hu, and K. Li, "A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues," Inf. Sci. (Ny)., vol. 270, pp. 255–287, Jun. 2014.
[28] M. Shojafar, M. Kardgar, A. A. R. Hosseinabadi, S. Shamshirband, and A. Abraham, "TETS: A Genetic-Based Scheduler in Cloud Computing to Decrease Energy and Makespan," in Hybrid Intelligent Systems, Springer, 2016, pp. 103–115.
[29] N. ; A. Y. Z. Young Choon Lee ; Adv. Networks Res. Group, Univ. of Sydney, Sydney, "Minimizing Energy Consumption for Precedence-Constrained Applications Using Dynamic Voltage Scaling," Clust. Comput. Grid, 2009. CCGRID '09. 9th IEEE/ACM Int. Symp., pp. 92–99.
[30] R. Min, T. Furrer, and A. Chandrakasan, "Dynamic voltage scaling techniques for distributed microsensor networks," in VLSI, 2000. Proceedings. IEEE Computer Society Workshop on, 2000, pp. 43–46.
[31] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," Adv. Eng. Softw., vol. 69, pp. 46–61, Mar. 2014.
[32] M. A. Tawfeek, A. El-Sisi, A. E. Keshk, and F. A. Torkey, "Cloud task scheduling based on ant colony optimization," in Computer Engineering & Systems (ICCES), 2013 8th International Conference on, 2013, pp. 64–69.
[33] M. Kalra and S. Singh, "A review of metaheuristic scheduling techniques in cloud computing," Egypt. Informatics J., vol. 16, pp. 275–295, 2015.
[8] W. Voorsluys, J. Broberg, and R. Buyya, "Introduction to cloud computing," Cloud Comput. Princ. Paradig., pp. 1–44, 2011.
[9] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Futur. Gener. Comput. Syst., vol. 25, no. 6, pp. 599–616, Jun. 2009.
[10] N. Jafari Navimipour, A. M. Rahmani, A. Habibizad Navin, and M. Hosseinzadeh, "Expert Cloud: A Cloud-based framework to share the knowledge and skills of human resources," Comput. Human Behav., vol. 46, pp. 57–74, May 2015.
[11] Y. Jadeja and K. Modi, "Cloud computing-concepts, architecture and challenges," in Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on, 2012, pp. 877–880.
[12] B. Keshanchi and N. J. Navimipour, "Priority-Based Task Scheduling in the Cloud Systems Using a Memetic Algorithm," J. Circuits, Syst. Comput., vol. 25, no. 10, p. 1650119, 2016.
[13] B. Keshanchi, A. Souri, and N. J. Navimipour, "An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: Formal verification, simulation, and statistical testing," J. Syst. Softw., vol. 124, pp. 1–21, 2017.
[14] M. Choudhary and S. K. Peddoju, "A dynamic optimization algorithm for task scheduling in cloud environment," Int. J. Eng. Res. Appl., vol. 2, no. 3, pp. 2564–2568, 2012.
[15] M. Habibi and N. J. Navimipour, "Multi-Objective Task Scheduling in Cloud Computing Using an Imperialist Competitive Algorithm," Int. J. Adv. Comput. Sci. Appl., vol. 1, no. 7, pp. 289–293, 2016.
[16] D. Mourad Oussalah and Professor Ali Hessami, N. Jafari Navimipour, A. Masoud Rahmani, A. Habibizad Navin, and M. Hosseinzadeh, "Job scheduling in the Expert Cloud based on genetic algorithms," Kybernetes, vol. 43, no. 8, pp. 1262–1275, 2014.
[17] N. J. Navimipour, "Task scheduling in the Cloud Environments based on an Artificial Bee Colony Algorithm," Proc. 2015 Int. Conf. Image Process. Prod. Comput. Sci.
[18] N. J. Navimipour and B. A. Milani, "Replica selection in the cloud environments using an ant colony algorithm," in Digital Information Processing, Data Mining, and Wireless Communications (DIPDMWC), 2016 Third International Conference on, 2016, pp. 105–110.
[19] H. Sun, S. Chen, C. Jin, and K. Guo, "Research and simulation of task scheduling algorithm in cloud computing," TELKOMNIKA Indones. J. Electr. Eng., vol. 11, no. 11, pp. 6664–6672, 2013.
[20] M. I. Daoud and N. Kharma, "A high performance algorithm for static task scheduling in heterogeneous distributed computing systems," J. Parallel Distrib. Comput., vol. 68, no. 4, pp. 399–409, 2008.
[21] C. Mateos, E. Pacini, and C. G. Garino, "An ACO-in-