ارزیابی کاهش مصرف انرژی با استفاده از پارامترهای شاخص در الگوریتم ژنتیک

آرش قربان نیا دلاور*[۱]، فاطمه علی بخشی[۲]

۱- عضو هیئت‌علمی، گروه مهندسی کامپیوتر و فناوری اطلاعات دانشگاه پیام نور تهران، ص .پ۰۲۶۵ ۲۶۳۶۷ ،. تهران، ایران

۲- دانشجوی کارشناسی ارشد، گروه مهندسی کامپیوتر و فناوری اطلاعات دانشگاه پیام نور تهران، تهران، ایران

چکیده:

رایانش ابری ارتباط بین user و سطح دسترسی زمانبند را در شرایط کمی و کیفی ارتقا می‌دهد به همین خاطر برای دسترسی سریع‌تر به داده‌ها بر روی ماشین‌های فیزیکی و پاسخگویی سریع‌تر به درخواست‌ها استفاده می‌شود. در این راستا در این مطالعه راه‌های بهبود سرعت و استفاده از سرویس‌های ابری ارزیابی شده است و با استفاده از آن Makespan و انرژی مصرفی کاهش داده شده است. کاهش انرژی مصرفی یادشده در بهبود نقطه محلی الگوریتم ژنتیک دارای تاثیر مستقیم بوده و کاهش انرژی مصرفی نقطه محلی را بهینه می‌کند. در سالهای اخیر استفاده از الگوریتم ژنتیک که یک رویکرد تکاملی می‌باشد برای انتخاب راه حل‌های بهتر در حل مسائل np-hard پیشنهاد شده است. پاسخگویی به درخواست‌ها و زمان‌بندی درست کارها منجر به کاهش زمان و مصرف انرژی در حل مسائل می‌شود. در نهایت با در نظر گرفتن پارامترهای شاخص نسبت به الگوریتم‌های FCFS,GfE,Detour,PSG,PSG-M، انرژی مصرفی ٪۱۲ کاهش داشته است و همچنین نیز زمان اجرای کل نسبت به روش‌های مورد مطالعه شده ٪۳ کاهش یافته است.

کلمات کلیدی: Energy consumption، Makespan، Workflow، Scheduling، Execution time

۱. مقدمه

روش سنتی صرفه‌جویی در انرژی در قرار دادن ماشین مجازی (VMP) مبتنی بر تجمیع ماشین‌های مجازی (VMs) بیشتر، در سرورهای کمتر و قرار دادن بقیه در حالت خواب است که ممکن است منجر به داغ شدن بیش از حد سرورها و در نتیجه کاهش عملکرد و افزایش هزینه خنک‌کننده شود. فقدان یک مدل دقیق و کارآمد محاسباتی برای توصیف شرایط حرارتی محیط مرکز داده، توسعه یک مکانیسم VMPموثر و سازگار را چالش‌برانگیز می‌کند[1].

با به حداکثر رساندن استفاده از منابع، GA پیشرفته برای یافتن گروهی از گره‌های سرویس که برای استقرار وظایف درخواستی استفاده می‌شوند، اجرا می‌شود. روش‌هایی مانند BAT، CSO، PSO، روش بهینه‌سازی CSA و WOA برای متعادل کردن بار، کارایی انرژی و بهبود زمان‌بندی منابع برای ایجاد یک پلتفرم ابری موثراست. ایده مجازی‌سازی به اجرای وظایف بسته به در دسترس بودن منابع کمک می‌کند و زمان پاسخگویی را کاهش می‌دهد[2].

زمان‌بندی کار با انرژی کارآمد با وظایف مستقل در یک DC منفرد بحث شده است، تعداد معینی از مقالات شامل زمان‌بندی جریان کار انرژی کارآمد با زمان‌های انتقال داده در GD-CDC ها است. به‌طورکلی، این الگوریتم‌های زمان‌بندی کار پیشنهادی را می‌توان به دو دسته تقسیم کرد: مبتنی بر اکتشاف و فراابتکاری. الگوریتم‌های زمان‌بندی مبتنی بر اکتشاف معمولاً یک استراتژی زمان‌بندی را بر اساس اطلاعات ناقص در زمان چندجمله‌ای پیدا می‌کنند. الگوریتم‌های زمان‌بندی فراابتکاری اغلب در روش‌های جستجوی تصادفی استفاده می‌شوند[3].

مشکل تخلیه وظایف مربوط به نحوه توزیع وظایف بارگیری شده توسط دستگاه‌های اینترنت اشیا در بین لایه‌های مه، ابر یا مه/ابر است تا اطمینان حاصل شود که زمان پردازش به حداقل می‌رسد .در اندازه‌گیری گره‌های مه، چالش بهینه‌سازی تعداد دستگاه‌های محاسباتی مه است که می‌توانند وظایف تعداد مشخصی از دستگاه‌های نهایی یا دستگاه‌های محاسباتی کمتر را پردازش کنند و در عین حال از کاهش زمان تکمیل کار و نرخ موفقیت بالاتر اطمینان حاصل کنند[4].

حجم کار در ماشین‌های مجازی در طول زمان متفاوت است .با وجود محدودیت چنین رویکردهایی تشدید سربار ناشی از مهاجرت بی‌مورد است که منجر به افزایش تاخیر و تنگناهای شبکه می‌شود .بنابراین، به حداقل رساندن تعداد مهاجرت‌هایVM ، استفاده کارآمد از منابع و تخصیص مجدد ماشین‌های مجازی موجود می‌تواند بهره‌وری سیستم را افزایش دهد. آن‌ها سرورهای ابری را به سمت دستگاه‌های سرور مجازی هدایت می‌کنند تا بهره‌وری انرژی سرورهای ابری را افزایش دهند[5] .

یک چارچوب سیستماتیک در جهت بهینه‌سازی اهداف برای اهداف مختلف مورد نیاز، با در نظر گرفتن حداکثر تاخیر، متعادل‌سازی بار و سازش با استفاده از یک الگوریتم مبتنی بر بازپخت شبیه‌سازی شده با هزینه از دست دادن دقت، حل می‌شود[6].

در تابع هزینه برای انجام تخصیص منابع ماهرانه، محدودیت‌های بعدی باید در نظر گرفته شود، یعنی مصرف برق باید همراه با کاهش نرخ داده کاهش یابد[7].

مصرف‌کنندگان سود بیشتری را از ذخیره‌سازی ابری می‌برند، زیرا آن‌ها تنها مسئول مقدار اطلاعاتی هستند که استفاده می‌کنند که در صورت درخواست می‌توان آن‌ها را افزایش و همچنین کاهش داد و هزینه‌های خرید و نگهداری زیرساخت داده را کاهش داد. با کاهش حجم اطلاعات، تامین کنندگان در واقع قادر خواهند بود از طریق استفاده از امکانات ذخیره‌سازی عظیم و همچنین کاهش مصرف انرژی در فناوری ابر، در هزینه خود صرفه‌جویی کنند. مشتری عملیات‌های متعددی را بر روی مجموعه‌های اطلاعاتی خاص به‌طور منظم انجام می‌دهد و این مجموعه داده‌ها باید برای امنیت و استفاده تصادفی تکرار شوند و در زمان و حافظه صرفه‌جویی شود. ایجاد یک تکراری تأخیر، دسترسی و همچنین استفاده از پهنای باند را کاهش می‌دهد که میانگین زمان اجرای کار را کاهش می‌دهد و استفاده از منابع ابری را بهبود می‌بخشد[8].

پس از شناسایی قوانین نظارت، قوانین را در یک مانیتور زمان اجرا ترکیب کردند که بررسی می‌کند آیا رفتار سیستم از رفتار مطلوب پیروی می‌کند یا خیر. هرگونه تخطی در زمان اجرا از قوانین نظارت، اخطار به موقع برای جلوگیری از عواقب نامطلوب، به عنوان مثال، خرابی‌های ثبت نشده، رفتار عدم توقف، انتشار خرابی در زیرسیستم ها و غیره می‌دهد[9].

هنگامی که ماشین‌های مجازی از طریق PM ها یا میزبان ساخته می‌شوند، هر ماشین مجازی از دو حالت به نام‌های بیکار و حالت فعال تشکیل می‌شود. به‌طور خاص، ارزیابی عملکرد شبکه ابری به‌طور کامل به عوامل حیاتی Makespan و مصرف انرژی (Energy Consumption) بستگی دارد. Makespan زمان اجرا را با توجه به هر ماشین مجازی در سیستم تعریف می‌کند. با این حال، زمان صرف شده برای تکمیل اجرای وظایف متفاوت است. LBبهتر به وضعیت به حداقل رساندن زمان اجرای وظایف کاربر ورودی اشاره دارد[10].

در ادامه این مطالعه در بخش۲ کارهای مرتبط با موضوع مطرح شده و در بخش ۳ تحلیل و ارزیابی کاهش مصرف انرژی انجام شده است. در بخش ۴ روش پیشنهادی ارائه شده است. بخش ۵ نتایج شبیه‌سازی را مورد بحث قرار داده و در نهایت در بخش ۶ نتیجه‌گیری ارائه شده است.

۲. کارهای پیشین:

H. Y. Shishido, J. C. (2017) تأثیر هر دو بهینه‌سازی ازدحام ذرات (PSO) و الگوریتم‌های مبتنی بر ژنتیک (GA)را بر تلاش‌ها برای بهینه‌سازی زمان‌بندی گردش کار بررسی می‌کند. برای ارزیابی عملکرد فراابتکاری، یک الگوریتم زمان‌بندی گردش کار آگاه از هزینه و امنیت انتخاب شد. سه الگوریتم در سه گردش کار دنیای واقعی با محدودیت نرخ ریسک، ارزیابی شدند. یافته‌ها نشان می‌دهد که الگوریتم‌های مبتنی بر GA به‌طور قابل‌توجهی از PSO هم از نظر اثربخشی هزینه و هم از نظر زمان پاسخ بهتر عمل می‌کند[11].

B. Keshanchi, A. S. (2016) برای بهینه‌سازی راه‌حل‌های زمان‌بندی کار، یک الگوریتم ژنتیک قدرتمند و بهبودیافته را پیشنهاد کرد. الگوریتم پیشنهادی از مزایای الگوریتم ژنتیک تکاملی همراه با رویکردهای اکتشافی استفاده می‌کند. زمان اجرای الگوریتم فراابتکاری آن‌ها عملکرد بهتری داشت[12].

(Yi Xie , et al.) الگوریتم ژنتیک کلید تصادفی را برای زمان‌بندی گردش کار ابری پیشنهاد کردند. در این الگوریتم، کدگذاری اعداد بر اساس کلید تصادفی با محدوده تعیین شده به کار گرفته شده است و برخی از طرح‌های جدید به عنوان مقداردهی اولیه جمعیت بر اساس سطح و اکتشافی از جمله اولین پایان ناهمگن پویا واقعی، بهبود یافته است. رمزگشایی تطبیقی پویا، تعادل بار با اجتناب از ارتباط و تکراری زمان به جلو، برنامه‌ریزی به عقب برای مقداردهی اولیه جمعیت، رمزگشایی کروموزوم طراحی شده است. زمان‌بندی در این الگوریتم‌ها نقش بسیار مهمی ایفا می‌کنند. یک زمان‌بندی درست موجب می‌شود که یک جریان داده با تاخیر کمتر و با کیفیت سرویس مناسب تحویل کاربران شود. این زمان‌بندی‌ها در مقاله ارائه شده بررسی و ارزیابی می‌شوند[13].

(Fatemeh Saadian , et al.) یک مدل بهینه‌سازی قرار دادن سرویس‌های برنامه‌های کاربردی آگاه از مهلت سه قسمتی در محیط مه را پیشنهاد می‌کنندکه مصرف برق کل، اتلاف کل منابع و تأخیر کل شبکه را به‌طور همزمان بهینه می‌کند. مدل پیشنهادی برنامه‌ها را در سه سطح بر اساس مهلت مربوط به آن‌ها اولویت‌بندی می‌کند و سپس مدل با استفاده از مدل موازی کاهش برازش اول و ترکیب الگوریتم ژنتیک حل می‌شود. نتایج شبیه‌سازی نشان‌دهنده برتری رویکرد پیشنهادی در برابر الگوریتم‌های مشابه از نظر کاهش مصرف انرژی، اتلاف منابع، تأخیر شبکه و نرخ رد سرویس است[14].

در(Muhammad Sohaib Ajmal , et al.) الگوریتم ژنتیک مورچه ترکیبی برای زمان‌بندی کار پیشنهاد شده است. الگوریتم پیشنهادی از ویژگی‌های الگوریتم ژنتیک و الگوریتم کلونی مورچه‌ها استفاده می‌کند و وظایف و ماشین‌های مجازی را به گروه‌های کوچک‌تر تقسیم می‌کند. پس از تخصیص وظایف، فرمون به ماشین‌های مجازی اضافه می‌شود. الگوریتم پیشنهادی به‌طور موثر فضای راه‌حل را با تقسیم وظایف به گروه‌ها و با شناسایی ماشین‌های مجازی بارگذاری شده کاهش می‌دهد. با توجه به حداقل فضای حل الگوریتم پیشنهادی، همگرایی و زمان پاسخ به‌طور قابل توجهی کاهش می‌یابد. این یک راه حل زمان‌بندی عملی برای به حداقل رساندن زمان اجرای گردش کار و وظایف پیدا می‌کند. الگوریتم پیشنهادی آن‌ها کاهش در زمان اجرا و هزینه‌های کلی مرکز داده را به دست آورد[15] .

در (Michael GuilhermeJordan , et al.) کوله‌پشتی چند بعدی ژنتیکی (GMK) برای بهینه‌سازی تخصیص چند هسته در محیط های CPU-FPGA توسط نویسندگان پیشنهاد شد. برای عملکرد (GMK-P) و انرژی-(GMK

(Eطراحی شده است. هر دو رویکرد دو مرحله را دنبال می‌کنند، ایجاد پیکربندی FPGA و بهینه‌سازی مشارکتی با استفاده از نسل پیکربندی FPGA شروع می‌شود، جایی که پیکربندی FPGA با رعایت محدودیت‌های منبع (LUT/FF/BRAM/DSP/IO)با استفاده از یک روش ژنتیکی که هسته‌هایی را انتخاب می‌کند. پس از دستیابی به شرط توقف عمل ژنتیکی، بهینه‌سازی مشارکتی را آغاز می‌کند. در این مرحله، هسته‌های باقی‌مانده را به CPU ارسال می‌کند تا پهنای ترتیب هسته افزایش نیابد. با این حال، GMK-E تنها زمانی هسته‌ها را به CPU اختصاص می‌دهد که انرژی کمتری نسبت به زمانی که در FPGA اجرا می‌شوند، مصرف کنند[16].

در (Sadoon Azizi , et al.) به صورت ریاضی مشکل زمان‌بندی کار را برای به حداقل رساندن مصرف انرژی کل گره‌های مه (FNs) در حالی که الزامات کیفیت خدمات (QoS) وظایف IoT را برآورده کردند، فرموله می‌کنند. آن‌ها همچنین به حداقل رساندن زمان نقض مهلت در مدل خود را در نظر می‌گیرند. در مرحله بعد، آن‌ها دو الگوریتم مبتنی بر نیمه حریص، یعنی نیمه حریص آگاه از اولویت (PSG) و PSG با رویه چند شروع (PSG-M) را برای ترسیم کارآمد وظایف اینترنت اشیا به FN ها پیشنهاد می‌کنند. عملکرد رویکردهای زمان‌بندی کار پیشنهادی را با توجه به درصد وظایف IoT که نیاز ضرب‌الاجل، مصرف کل انرژی، کل زمان نقض مهلت و طول عمر سیستم را برآورده می‌کنند، ارزیابی می‌نمایند. در مقایسه با الگوریتم‌های موجود، نتایج آزمایش تأیید می‌کند که الگوریتم‌های پیشنهادی درصد وظایفی را که نیاز به ضرب‌الاجل خود را برآورده می‌کنند تا ۱.۳۵ برابر بهبود می‌بخشد و زمان کل نقض مهلت را تا ۹۷.۶ درصد نسبت به دومین نتایج برتر کاهش می‌دهد، در حالی که انرژی مصرف منابع مه و ساختار سیستم بهینه‌شده است.. آن‌ها ابتدا مسئله زمان‌بندی کار را به‌عنوان یک برنامه‌ریزی غیرخطی عدد صحیح مختلط (MINLP) با هدف به حداقل رساندن مصرف انرژی و محدودیت رعایت ضرب‌الاجل وظایف اینترنت اشیا مدل‌سازی نمودند. همچنین به حداقل رساندن زمان نقض مهلت در مدل خود را در نظر می‌گیرند. برای ارائه یک راه‌حل کارآمد برای هر نمونه از مشکل، دو الگوریتم مبتنی بر نیمه حریص به نام‌های نیمه حریص آگاه از اولویت (PSG) و PSG با روش چند شروع (PSG-M) پیشنهاد کردند. آن‌ها شبیه‌سازی گسترده‌ای را برای تأیید اثربخشی الگوریتم‌های پیشنهادی انجام دادند. پیش از این، دو الگوریتم اکتشافی حریصانه را برای حل مشکل زمان‌بندی کار در سیستم‌های محاسباتی داوطلبانه (VCSs) با تمرکز بر به حداقل رساندن هزینه کل از نظر محاسبات، ارتباطات و نقض مهلت پیشنهاد کرده بودند[17].

ما در این مطالعه انرژی مصرفی و Makespanرا به‌طور همزمان نسبت به روش‌های مطالعه شده کاهش داده‌ایم.

۳. تحلیل و ارزیابی کاهش مصرف انرژی:

امروزه رایانش ابری یکی از مهم‌ترین فناوری‌ها می‌باشد و باعث ایجاد تحولات عظیمی در زمینه فناوری اطلاعات شده است. استفاده از این فناوری به دلیل سرعت ارائه خدمات به مشتریان و کاهش هزینه‌ها مورد توجه قرار گرفته است. با توجه به این موضوع، رفع کاستی‌های این فناوری باعث انجام مطالعات بسیاری شده است.

ابر یک الگوی ذخیره‌سازی کامپیوتری است که در آن اطلاعات دیجیتال در گروه‌های مجازی سازمان‌دهی می‌شود. یک کسب‌وکار میزبانی اغلب دنیای فیزیکی را تحت کنترل دارد و چندین سرور را در بر می‌گیرد. ارائه دهندگان ذخیره‌سازی ابری، مسئول حفظ امنیت و در دسترس نگه‌داشتن اطلاعات و محیط فیزیکی امن و عملیاتی هستند. افراد و سازمان‌ها قابلیت‌های ذخیره‌سازی را از ارائه‌دهندگان خدمات ابری برای ذخیره داده‌های مشتری، سازمان یا برنامه‌ها خریداری یا اجاره می‌کنند. ذخیره‌سازی ابری به دلیل ظرفیت آن در ارائه خدمات ذخیره‌سازی داده و قابل اعتماد بودن آن به‌طور فزاینده‌ای در بین مشاغل محبوب شده است. افزایش تقاضای نرم‌افزار به عنوان یک سرویس، باعث افزایش زیرساخت‌های انبارها شده

و مصرف انرژی را بالا برده است. حجم کاری بسیار متنوع نیاز به بهینه‌سازی همزمان چندین درخواست هسته (قطعات برنامه خاص که ممکن است بر روی CPU اجرا شود) از برنامه‌های مختلف دارد.

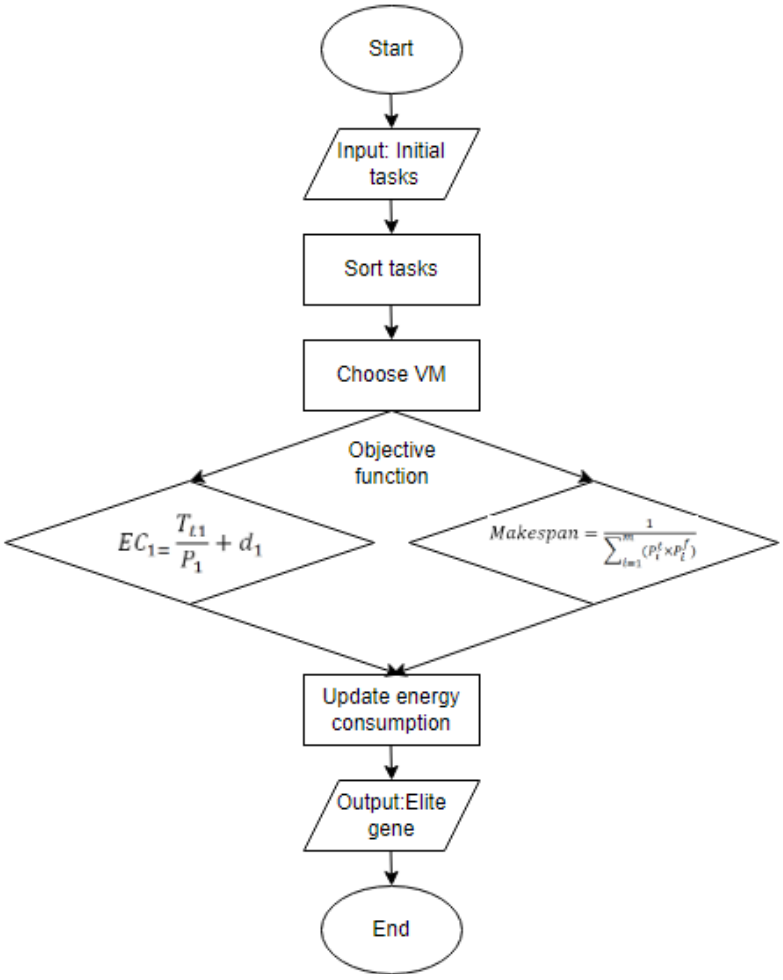
تکثیر داده‌ها یک رویکرد بسیار خوب است که شامل نگهداری کپی‌های متعددی از داده‌ها، به نام کپی، در چندین مکان است. اهداف آن بهبود در دسترس بودن داده‌ها، کاهش مصرف پهنای باند و بهبود تحمل خطا است. تکثیر داده‌ها به‌طور گسترده در فرآیندهای سنتی مانند P2P و فناوری‌های شبکه داده استفاده می‌شود. اما ساختن تعداد زیادی کپی در سیستم‌های ابری مقرون به صرفه نیست، زیرا منابع را هدر می‌دهد و سودآوری ارائه‌دهنده را به حداقل می‌رساند.

محاسبات ابری سعی دارد پتانسیل اشتراک منابع محاسباتی را بین کاربران مختلف در پلتفرم داشته باشد. در زمینه توسعه مراکز داده، ایجاد برخی مراکز تعیین شده که برنامه‌ها را مستقل از زیرساخت سخت‌افزاری اجرا می‌کنند، ضروری است. این پلتفرم همچنین امکان انتقال منابع از یک ماشین مجازی (VM) به دیگری را برای پشتیبانی از خدمات یکپارچه در بین ماشین‌های مجازی منبع و وظایف فراهم می‌کند. در محاسبات ابری، برنامه‌ریزی بهینه وظایف، مدیریت مصرف انرژی، LBو تخصیص منابع به عنوان مهم‌ترین چالش‌ها و موضوعاتی در نظر گرفته می‌شوند که برای استفاده کارآمد از محیط ابری باید متمرکز شوند. در این میان به حداقل رساندن مصرف انرژی و همچنین به حداقل رساندن زمان پاسخ بسیار مهم است.

بعضی از تکنیک‌های مدیریت توانCPU ، مانند DVFS (مقیاس ولتاژ و فرکانس دینامیک)، می‌تواند برای بهره‌وری انرژی استفاده شود. این تکنیک از بیکاری پردازنده (معمولاً به دلیل عملیات ورودی/خروجی) برای کاهش در زمان اجرا، فرکانس کاری و ولتاژ منبع بهره می‌برد. مقیاس بندی ولتاژ و فرکانس دینامیکی (DVFS) فرکانس کاری یک پردازنده را کاهش می‌دهد. در نتیجه، مصرف برق کاهش می‌یابد و منجر به صرفه‌جویی قابل توجهی در انرژی می‌شود.

بهینه‌سازی تخصیص منابع در شبکه چالش‌های قابل توجهی مانند وقت‌گیر بودن را شامل می‌شود. در تخصیص منابع، کل درجات آزادی در نظر گرفته می‌شود و پس از آن تعداد دلخواه کاربران مجاز به تسهیم کننده شدن در یک زیر حامل مشابه است.

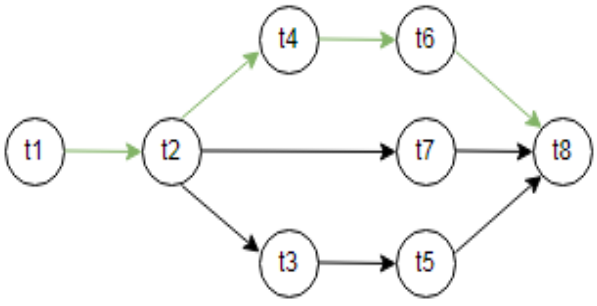
ما در این مطالعه به ارزیابی مصرف انرژی و زمان اجرای کل با استفاده از الگوریتم ژنتیک می‌پردازیم. Makespan یا زمان اجرای کل، زمان صرف شده برای پردازش مجموعه‌ای از کارها برای اجرای کامل آن‌ها است و انرژی مصرفی (Energy Consumption) انرژی مصرف‌شده در طول اجرای درخواست‌ها بر روی یک منبع است. شکل ۱ فلوچارت مصرف انرژی را نشان می‌دهد.



شکل۱- فلوچارت طرح پیشنهادی

۴. طرح پیشنهادی

همانطور که در شکل۲ نشان داده شده یک گردش کار را می‌توان به‌صورت یک گراف جهت‌دار در نظر گرفت که وظایف به‌صورت $Tasks=\{t_1,t_2,t_3,\ldots,t_n\}$ و یال‌ها به‌صورت $E=\{e_1,e_2,e_3,\ldots,e_4\}$، نمایش داده می‌شوند؛ و منابع را می‌توان با مجموعه $VMs=\{v_1,v_2,v_3,\ldots,v_n\}$، نمایش داد. مسیر بهینه در این گراف $\{t1,t2,t4,t6,t8\}$ می‌باشد که با فلش سبز رنگ نشان داده شده است.



شکل۲- نمونه ای از یک گردش کار

هر یک از taskها نشان‌دهنده‌ی درخواست‌های ارسالی توسط کاربران است؛ که ما آن‌ها را یک ژن در نظر گرفته و پاسخگویی به این درخواست‌ها بر اساس اهمیت آن‌ها و همچنین زمان پاسخگویی به آن‌ها یکی از چالش‌های این مقاله بوده که با استفاده از الگوریتم ژنتیک بررسی می‌شود. ما task های با اولویت بالاتر را به عنوان ژن‌های برتر در نظر گرفته‌ایم. با توجه به اینکه ما در الگوریتم ژنتیک ژن‌های برتر را برای تولید نسل‌های بهتر انتخاب می‌کنیم، با الگوبرداری از این الگوریتم taskها را انتخاب نموده و در ابتدا به taskهای با اولویت بالاتر پاسخ می‌دهیم و با استفاده از الگوریتم ژنتیک از تکرار نسل جلوگیری نموده که این امر انرژی مصرفی و Makespan را کاهش می‌دهد. یکی از ایرادات الگوریتم ژنتیک نقطه محلی می‌باشد ما با کاهش انرژی مصرفی نقطه محلی را بهینه نموده‌ایم. جدول ۱ پارامترها را نشان می‌دهد.

جدول۱- پارامترهای استفاده شده

Parameters	
T	**Task**
E_{tot}	**Total energy**
EC	**Energy consumotion**
RFT	**Real free time**
Exe	**Execution time**
ER	**Execution real time**
TS	**Time spent**
M	**Makespan**
d	**distance**

همانطور که میدانیم انرژی در ابتدای پروسه پردازش بیشتر بوده و به مرور با بررسی و پاسخگویی به درخواست‌های کاربران انرژی مصرفی و انرژی تخلیه شده افزایش میابد. رابطه (۱) انرژی آزاد شده را نشان می‌دهد؛ و رابطه (۲) انرژی کل را نشان می‌دهد. انرژی هر task از رابطه (۱) محاسبه می‌شود:

$$EC_1 = \frac{T_{t1}}{P_1} + d_1 \qquad (۱)$$

که در آن EC_1 انرژی مصرف شده برای $task_1$ و T_{t1} زمان اجرای $task_1$ بر روی vm_1 و P توان پردازشی می‌باشد، و به همین ترتیب برای بقیه task ها $\{t_1, t_2, t_3, \ldots, t_n\}$، انرژی به این صورت محاسبه شده و انرژی کل از رابطه(۲) محاسبه می‌شود.

$$E_{tot} = \sum_{i=1}^{n} \mathrm{EC} \qquad (۲)$$

و همچنین می‌توان تعداد انرژی‌های مصرفی را از رابطه (۳) به دست آورد.

$$N = \frac{AVG\ EC_{1\ldots n}}{AVG\ E_{tot}} \qquad (۳)$$

تابع Fitness نیز از رابطه(۴) بدست می‌آید.

$$Fitness = \sum_{i=1}^{m} (P_i^t \times P_i^f) \qquad (٤)$$

که در آن P_i^t زمانی است که منبع P_i تمام وظایف محول شده گردش کار را در زمان‌بندی جاری تکمیل می‌کند. در این تابع کل زمان صرف شده هر منبع برای کاربرد فعلی در نظر گرفته می‌شود. که بر روی Makespan موثر است. همچنین P_i^f را در نظر می‌گیریم تا میزان شکست در اجرای وظایف را تا حد ممکن کاهش دهیم[18].

و Makespan از رابطه (۵) به دست می‌آید.

$$Makespan = \frac{1}{\sum_{i=1}^{m}\left(P_i^t \times P_i^f\right)} \quad (۵)$$

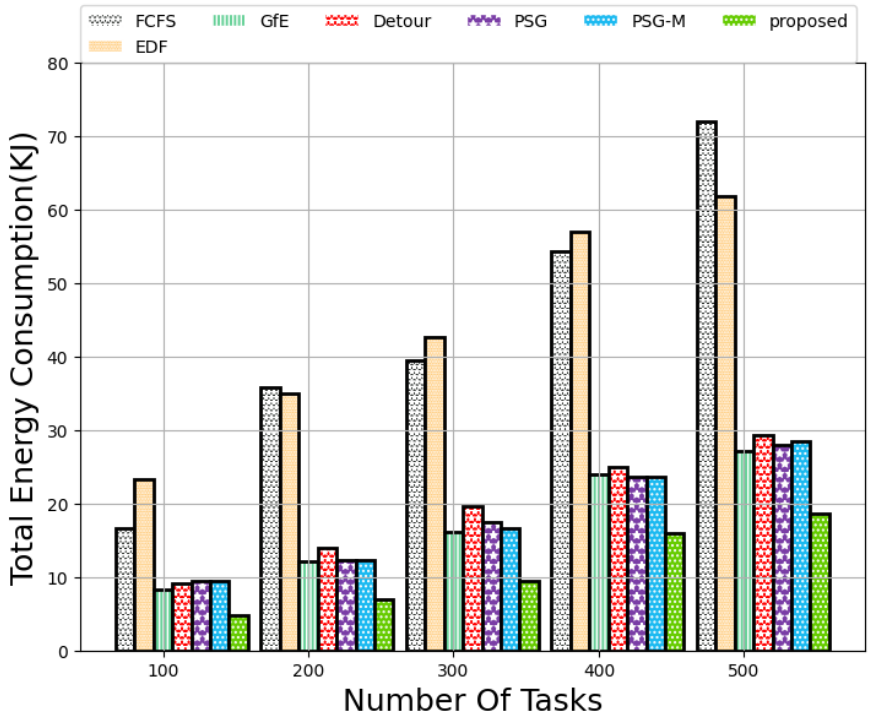
۵. نتایج تجربی

در جدول ۲ نتایج مقایسه عددی هفت الگوریتم نشان داده شده است. همانطور که در جدول مشاهده می‌شود با بالا رفتن تعداد taskها مصرف انرژی بالا می‌رود و برتری روش ما (Proposed) از نظر مصرف انرژی به وضوح مشخص می‌باشد.

جدول ۲: مقایسه مصرف انرژی بین هفت الگوریتم

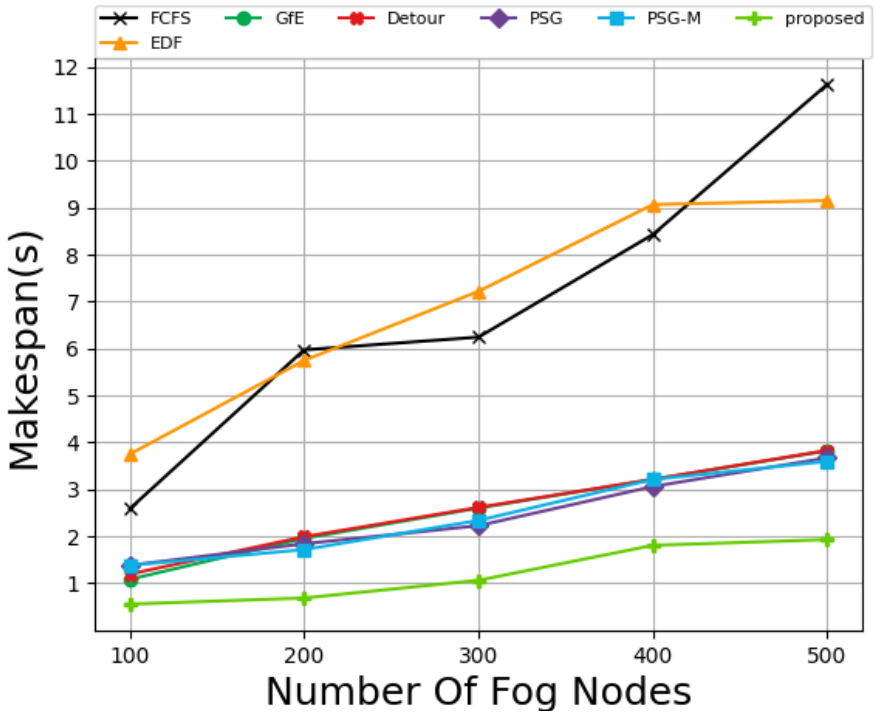
Number of tasks	FCFS	EDF	GfE	Detour	PSG	PSG-M	Proposed
100	16.6443	23.3557	8.32215	9.12752	9.39597	9.39597	4.832216
200	35.7047	34.8993	12.0805	13.9597	12.349	12.349	6.92618
300	39.4631	42.6846	16.1074	19.5973	17.4497	16.6443	9.39598
400	54.2282	56.9128	23.8926	24.9664	23.6242	23.6242	16.00004
500	71.9463	61.745	27.1141	29.2617	27.9195	28.4564	18.68462

شکل ۴ نتایج شبیه‌سازی هفت الگوریتم با تعدادtaskهای متغیر بین ۱۰۰ تا ۵۰۰ task و انرژی مصرفی را در محدوده ۰ تا ۸۰ نشان می‌دهد و نتایج بهبودی انرژی مصرفی را نسبت سایر الگوریتم‌ها نشان داده است.



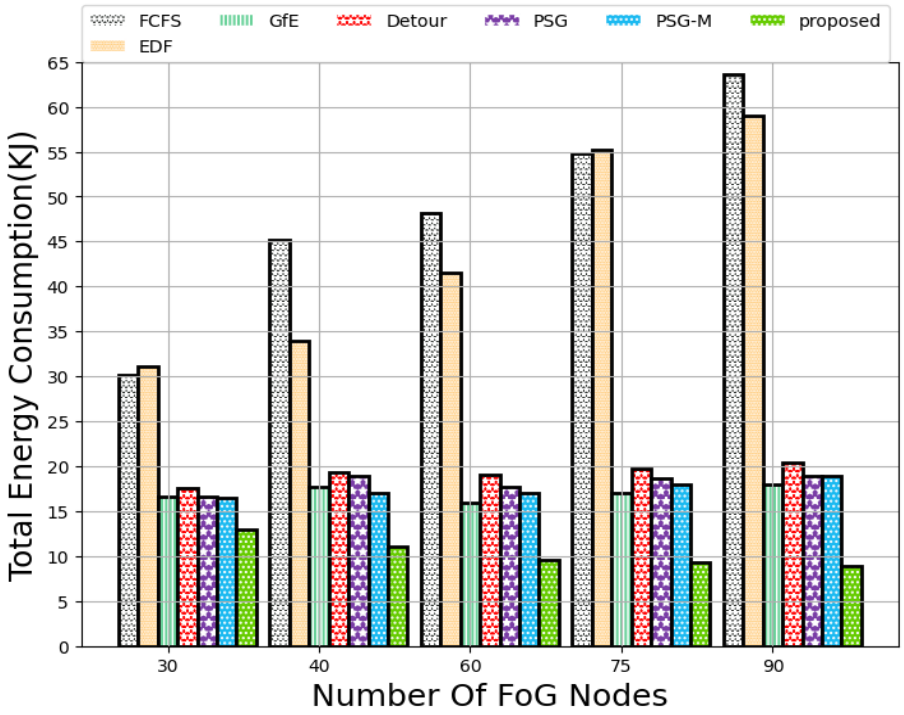
شکل۴: نتایج شبیه‌سازی مصرف انرژی در الگوریتم‌ها

شکل۵ زمان اجرا کل را در محدوده۱۰۰ تا ۵۰۰ گره و زمان اجرا را در محدوده ۰ تا ۱۲ثانیه مقایسه نموده که نتایج نشان‌دهنده بهبود Makespan در روش پیشنهادی نسبت به سایر الگوریتم‌ها و برتری آن می‌باشد.



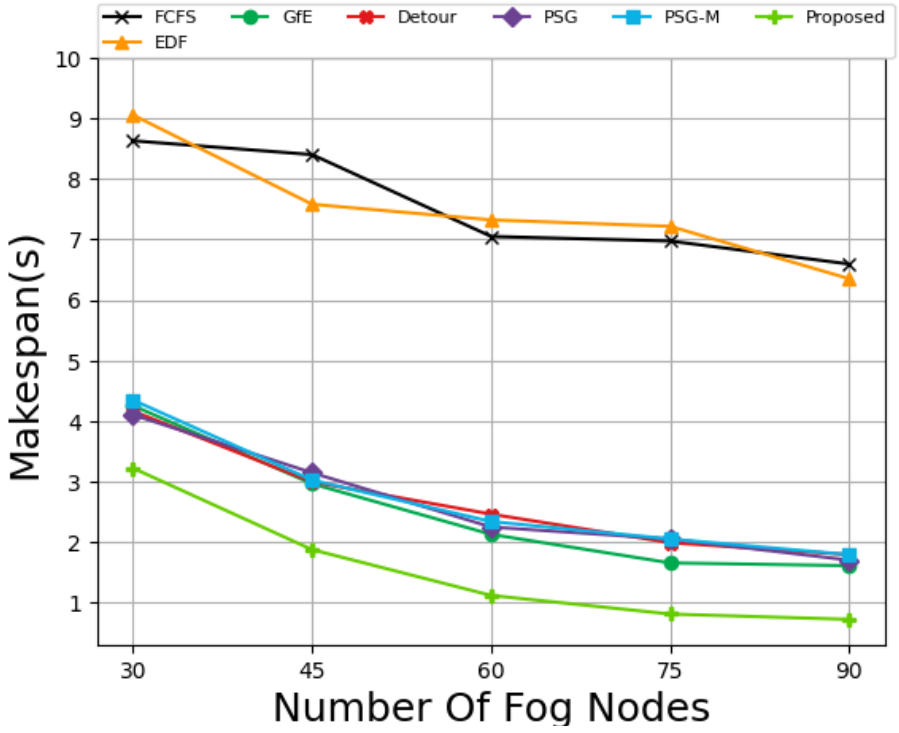
شکل۵: نتایج شبیه‌سازی Makespan در الگوریتم‌ها

شکل ۶ نتایج مقایسه الگوریتم از نظر مصرف انرژی بر روی تعداد متغیر Taskها (بین ۳۰ تا ۹۰ Task) و محدوده انرژی بین ۰ تا ۶۵ را نشان می‌دهد.



شکل6: نتایج شبیه‌سازی مصرف انرژی بر روی گره‌ها

شکل۷، Makespan را بر روی تعداد گره‌های متغیر (بین ۳۰ تا ۹۰ گره) و زمان اجرا را در محدوده۰ تا ۱۰ ثانیه نشان می‌دهد که در هر دو شکل انرژی مصرفی و زمان اجرای کل بهبود یافته‌اند.



شکل6: نتایج شبیه‌سازی makespan بر روی گره‌ها

۶. نتیجه‌گیری

با توجه به استفاده روزافزون از رایانش ابری برطرف کردن چالش‌های این فناوری در زمینه‌های مختلف مطرح می‌شود. از چالش‌های مهم محاسبات ابری کاهش مصرف انرژی و همچنین کاهش زمان اجرای کل (Makespan) می‌باشد. با در نظر گرفتن موارد ذکر شده پژوهشی پیشنهاد شد که مصرف انرژی را به‌طورکمینه و بیشینه تا٪۴ و ٪۱۲کاهش داده است و به‌طور میانگین ٪ ۱۱ کاهش داشته است. همچنین زمان اجرای کل را در کمینه و بیشینه٪۰.۵ و٪۳ درصد کاهش داشته و به‌طور میانگین٪ ۱.۳ کاهش یافته است.

۷. مراجع

[1] A. Aghasi, K. Jamshidi, A. Bohlooli, and B. Javadi, “A decentralized adaptation of model-free Q-learning for thermal-aware energy-efficient virtual machine placement in cloud data centers,” *Comput. Networks*, vol. 224, Apr. 2023, doi: 10.1016/j.comnet.2023.109624.

[2] F. N. Al-Wesabi, M. Obayya, M. A. Hamza, J. S. Alzahrani, D. Gupta, and S. Kumar, “Energy Aware Resource Optimization using Unified Metaheuristic Optimization Algorithm Allocation for Cloud Computing Environment,” *Sustain. Comput. Informatics Syst.*, vol. 35, Sep. 2022, doi: 10.1016/j.suscom.2022.100686.

[3] M. Hussain, L. F. Wei, A. Rehman, F. Abbas, A. Hussain, and M. Ali, “Deadline-constrained energy-aware workflow scheduling in geographically distributed cloud data centers,” *Futur. Gener. Comput. Syst.*, vol. 132, pp. 211–222, Jul. 2022, doi: 10.1016/j.future.2022.02.018.

[4] S. O. Ogundoyin and I. A. Kamil, “Optimal fog node selection based on hybrid particle swarm optimization and firefly algorithm in dynamic fog computing services,” *Eng. Appl. Artif. Intell.*, vol. 121, May 2023, doi: 10.1016/j.engappai.2023.105998.

[5] M. I. Khaleel, “Synergies between resource sustainability and energy performance of cloud servers: The role of virtual machine repacking approach,” *Comput. Electr. Eng.*, vol. 106, Mar. 2023, doi: 10.1016/j.compeleceng.2022.108568.

[6] C. Li, K. Jiang, and Y. Luo, “Dynamic placement of multiple controllers based on SDN and allocation of computational resources based on heuristic ant colony algorithm,” *Knowledge-Based Syst.*, vol. 241, Apr. 2022, doi: 10.1016/j.knosys.2022.108330.

[7] B. V. L. A. and A. D. G., “Hybrid optimization using lion and dragonfly for enhanced resource allocation in fifth-generation networks,” *Data Knowl. Eng.*, vol. 145, May 2023, doi: 10.1016/j.datak.2023.102151.

[8] R. D and G. A, “Optimization assisted frequent pattern mining for data replication in cloud: Combining sealion and grey wolf algorithm,” *Adv. Eng. Softw.*, vol. 176, Feb. 2023, doi: 10.1016/j.advengsoft.2022.103401.

[9] D. Cotroneo, L. De Simone, P. Liguori, and R. Natella, “Run-time failure detection via non-intrusive event analysis in a large-scale cloud computing platform,” *J. Syst. Softw.*, vol. 198, Apr. 2023, doi: 10.1016/j.jss.2023.111611.

[10] S. Janakiraman and M. D. Priya, “Hybrid grey wolf and improved particle swarm optimization with adaptive intertial weight-based multi-dimensional learning strategy for load balancing in cloud environments,” *Sustain. Comput. Informatics Syst.*, vol. 38, Apr. 2023, doi: 10.1016/j.suscom.2023.100875.

[11] H. Y. Shishido, J. C. Estrella, C. F. M. Toledo, and M. S. Arantes, “Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds,” *Comput. Electr. Eng.*, vol. 69, pp. 378–394, Jul. 2018, doi: 10.1016/J.COMPELECENG.2017.12.004.

[12] 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, 2017, doi: 10.1016/j.jss.2016.07.006.

[13] Y. Xie, Y. Sheng, M. Qiu, and F. Gui, “An adaptive decoding biased random key genetic algorithm for cloud workflow scheduling,” *Eng. Appl. Artif. Intell.*, vol. 112, Jun. 2022, doi: 10.1016/j.engappai.2022.104879.

[14] F. Saadian, H. Motameni, and M. Golsorkhtabaramiri, “Deadline-aware multi-objective IoT services placement optimization in fog environment using parallel FFD-genetic algorithm,” *Pervasive Mob. Comput.*, vol. 92, May 2023, doi: 10.1016/j.pmcj.2023.101800.

[15] M. S. Ajmal, Z. Iqbal, F. Z. Khan, M. Ahmad, I. Ahmad, and B. B. Gupta, “Hybrid ant genetic algorithm for efficient task scheduling in cloud data centers,” *Comput. Electr. Eng.*, vol. 95, Oct. 2021, doi: 10.1016/j.compeleceng.2021.107419.

[16] M. G. Jordan, G. Korol, T. Knorst, M. B. Rutzig, and A. C. S. Beck, “Energy-aware fully-adaptive resource provisioning in collaborative CPU-FPGA cloud environments,” *J. Parallel Distrib. Comput.*, vol. 176, pp. 55–69, 2023, doi: 10.1016/j.jpdc.2023.02.009.

[17] S. Azizi, M. Shojafar, J. Abawajy, and R. Buyya, “Deadline-aware and energy-efficient IoT task scheduling in fog computing systems: A semi-greedy approach,” *J. Netw. Comput. Appl.*, vol. 201, May 2022, doi: 10.1016/j.jnca.2022.103333.

[18] A. Ghorbannia Delavar and Y. Aryan, “HSGA: A hybrid heuristic algorithm for workflow scheduling in cloud systems,” *Cluster Comput.*, vol. 17, no. 1, pp. 129–137, 2014, doi: 10.1007/s10586-013-0275-6.