

جای گذاری بهینه ماشین های مجازی در مراکز داده ابری بر پایه ترکیب الگوریتم جمعیت سالپ چند جمعیتی گسسته و الگوریتم تپه نوردی

محمد مصدري*^(۲) احمد جعفریان^(۳) ساسان قره پاشا^(۱)

(۱) گروه مهندسی کامپیوتر، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران

(۲) گروه مهندسی کامپیوتر، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران*

(۳) گروه ریاضی، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران

تاریخ پذیرش: ۱۳۹۹/۱۲/۲۶

تاریخ دریافت: ۱۳۹۸/۱۲/۲۶

چکیده

محاسبات ابری یک سرویس محاسباتی بوده که از طریق بستر اینترنت می توان به آن دسترسی داشت. منابع محاسباتی به اشتراک گذاری شده در رایانش ابری برای استفاده از طریق اینترنت در اختیار کاربران برای استفاده، قرار می گیرد. استفاده از تکنولوژی مجازی سازی در منابع محاسباتی این امکان را در رایانش ابری فراهم می آورد که از منابع به صورت مناسب استفاده شود. یکی از مهمترین موضوعات مورد چالش در تکنولوژی مجازی سازی، جای گذاری بهینه ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده ابری می باشد. جای گذاری بهینه ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده ابری باعث مدیریت مصرف منابع شده و از اتلاف آن جلوگیری می کند. در این مقاله، ما یک راهکار جدید را با استفاده از ترکیب الگوریتم فرا ابتکاری جمعیت سالپ چند جمعیتی به صورت گسسته و الگوریتم تپه نوردی برای جای گذاری بهینه ارائه داده ایم. اولین هدف از الگوریتم پیشنهادی، کاهش مصرف انرژی در مراکز داده ابری به وسیله کاهش ماشین های فیزیکی فعال می باشد. دومین هدف، کاهش اتلاف منابع و مدیریت آن به وسیله جای گذاری بهینه ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده ابری می باشد. با استفاده از این روش، از افزایش مهاجرت ماشین های مجازی بر روی ماشین های فیزیکی جلوگیری می شود. در نهایت، الگوریتم پیشنهادی با الگوریتم های کار شده در این زمینه از جمله GA، ACO، FF، GSA و FCFS مقایسه می شود.

واژه های کلیدی: الگوریتم جمعیت سالپ، الگوریتم تپه نوردی، رایانش ابری، مجازی سازی، چند جمعیتی

*عده دار مکاتبات :

محمد مصدري

نشانی: گروه مهندسی کامپیوتر، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران

پست الکترونیک : m.masdari@iaurmia.ac.ir : تلفن : ۰۹۱۴۳۸۹۹۰۰۸

۱- مقدمه

محاسبات ابری یک سرویس محاسباتی بوده که از طریق بستر اینترنت می‌توان به آن دسترسی داشت. منابع محاسباتی به اشتراک‌گذاری شده در رایانش ابری برای استفاده از طریق اینترنت در اختیار کاربران برای استفاده، قرار می‌گیرد. رایانش ابری به یک سرویس همگانی تبدیل شده است و ترکیبی از مفاهیم محاسباتی موازی و توزیع شده می‌باشد. در رایانش ابری، منابع محاسباتی از جمله سرور، منابع ذخیره‌سازی، شبکه، نرم‌افزارها و غیره به عنوان سرویس برای کاربران ابری تعریف شده است. منابع محاسباتی در یک مکان به نام مرکز داده ابری تعبیه شده‌اند و این مکان به وسیله سازمانی به نام تولیدکننده سرویس ابری مدیریت می‌شود. این سازمان سه نوع سرویس به نام‌های ساختار به عنوان سرویس، نرم‌افزار به عنوان سرویس و پلتفرم به عنوان سرویس را برای کاربران ابری ارائه می‌کند [۱-۳]. رایانش ابری دارای چهار نوع مدل ابری به نام‌های ابرهای عمومی، ابرهای ارتباطی، ابرهای خصوصی و ابرهای ترکیبی می‌باشد [۴-۵]. از بسترهای ابری می‌توان به آمازون، موتور محاسباتی گوگل و غیره اشاره نمود. محاسبات ابری فواید فراوانی را برای کاربران خود از جمله انعطاف‌پذیری، کاهش هزینه، وابستگی سخت‌افزاری و اعتماد فراهم کرده است [۶-۱۰]. در مرکز داده ابری، برای اجرای درخواست کاربر، ماشین‌های مجازی به صورت پویا ساخته شده یا از ماشین‌های مجازی از قبل تعریف شده استفاده می‌کنند. هر ماشین مجازی که بر روی ماشین فیزیکی قرار می‌گیرد نیازمند یکسری منابع فیزیکی بوده که به وسیله ماشین فیزیکی میزبان تامین می‌گردد. در این زمان، موارد کلیدی و مهم در بستر ابری به حداقل رساندن

میزان مصرف انرژی و اتلاف منابع در مراکز داده ابری می‌باشد. یکی از راهکارها برای رسیدن به اهداف بالا، استفاده از الگوریتم‌های مناسب جای‌گذاری ماشین‌های مجازی می‌باشد. وظیفه اصلی الگوریتم‌های جای‌گذاری ماشین‌های مجازی، مکان‌یابی ماشین‌های مجازی بر روی ماشین‌های فیزیکی به صورتی که استفاده از منابع محاسباتی مناسب بوده و از اتلاف آن جلوگیری شود و همچنین تعداد ماشین‌های فیزیکی فعال در مرکز داده ابری به حداقل برسد. چندین الگوریتم جای‌گذاری ماشین‌های مجازی در سال‌های اخیر ارائه شده است [۱۱-۲۲]. این الگوریتم‌ها توانایی ایجاد توازن مناسب در هنگام استفاده از منابع محاسباتی و همچنین جلوگیری از اتلاف منابع و به حداقل رساندن مصرف انرژی به صورت هم‌زمان را ندارند. در این مقاله، ما یک الگوریتم جدید برای جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی برای مدیریت بهتر منابع محاسباتی و همچنین مصرف انرژی ارائه می‌دهیم. اولین هدف از الگوریتم پیشنهادی، کاهش مصرف انرژی در مراکز داده ابری به وسیله کاهش ماشین‌های فیزیکی فعال می‌باشد. دومین هدف، کاهش اتلاف منابع و مدیریت آن به وسیله جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری می‌باشد. همچنین موارد دیگر از جمله کاهش ترافیک و کاهش میزان تاخیر مورد استفاده در مرکز داده ابری مورد بررسی قرار خواهد گرفت. استفاده از راهکار پیشنهادی باعث کاهش تعداد مهاجرت‌ها در مرکز داده ابری می‌شود. الگوریتم پیشنهادی با الگوریتم‌های کار شده در این زمینه از جمله GA، ACO، FF، GSA و FCFS مقایسه می‌شود. ادامه مقاله بدین صورت است که در بخش دوم برخی از پیشینه‌های تحقیق،

در بخش سوم مسئله جای‌گذاری ماشین‌های مجازی و در قسمت چهارم مفاهیم پایه مورد بحث قرار می‌گیرد. در بخش پنجم الگوریتم پیشنهادی ارائه می‌شود و در نهایت در بخش ششم آزمایش‌ها و نتایج حاصل از آن و در بخش هفتم نتیجه‌گیری حاصل از انجام این تحقیق مطرح می‌گردد.

۲- مروری بر کارهای پیشین

انواع متنوعی از الگوریتم‌های جای‌گذاری ماشین‌های مجازی در راینش ابری وجود دارد. هدف اصلی الگوریتم‌های جای‌گذاری ماشین‌های مجازی به دست آوردن عملکرد بالا و مدیریت منابع سیستم است. الگوریتم‌های جای‌گذاری سنتی قادر به فراهم کردن شرایط مناسب در محیط ابر نیستند زیرا سربار هزینه دارند و فراهم‌کنندگان به سمت استفاده از الگوریتم‌های اکتشافی یا ترکیبی می‌روند. در این بخش به بررسی قسمتی از کارهای انجام شده در این زمینه پرداخته می‌شود.

ساتپاتی و همکاران در سال ۲۰۱۷ یک روش جدید برای حل مسئله جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مرکز داده ابری ارائه داده‌اند. روش ارائه شده با استفاده از الگوریتم فرا ابتکاری کلاغ، عملیات جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی را در مراکز داده ابری انجام می‌دهد. با توجه به تعداد زیاد ماشین‌های مجازی نیاز به بهینه‌ترین روش برای جای‌گذاری آنها داریم. اهداف اصلی این روش به حداقل رساندن مصرف انرژی و اتلاف منابع در مراکز داده ابری و بالا بردن کارایی آن مراکز می‌باشد. جای‌گذاری بهینه ماشین‌های مجازی باعث کاهش تعداد ماشین‌های فیزیکی فعال شده و اهداف الگوریتم ارائه شده را بر آورده می‌کند. نتایج به دست آمده توسط

الگوریتم پیشنهادی با نتایج الگوریتم First Fit مقایسه شده و دارای بهبود بیشتری می‌باشد [۲۳].

رضانی و همکاران در سال ۲۰۱۶ یک الگوریتم جای‌گذاری جدید برای حل مسئله جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری ارائه کرده‌اند. الگوریتم مورد استفاده در روش پیشنهادی، الگوریتم بهینه‌سازی ازدحام ذرات می‌باشد. مدیریت ماشین‌های مجازی، نحوه قرار گرفتن ماشین‌های مجازی بر روی ماشین‌های فیزیکی و مدیریت مهاجرت ماشین‌های مجازی در مراکز ابری باعث استفاده بهینه از منابع سخت‌افزاری می‌شود. وجود تکنیک مهاجرت ماشین‌های مجازی در مراکز ابری تاثیر بسیاری در مدیریت مفید منابع دارد. الگوریتم ارائه شده که با تکنیک منطبق فازی بهبود یافته است باعث مصرف بهینه انرژی و همچنین افزایش کیفیت سرویس در مراکز ابری شده است. نتایج مقایسه شده نشان دهنده این می‌باشد که روش پیشنهادی دارای عملکرد بهتری نسبت به سایر روش‌های مقایسه شده می‌باشد [۲۴].

مولامی و همکاران در سال ۲۰۱۹ الگوریتم فرا ابتکاری بر پایه الگوریتم بهینه‌سازی گرگ خاکستری برای حل مسئله جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری ارائه کرده‌اند. مسئله جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری یکی از مهم‌ترین مسائل می‌باشد. بهینه‌سازی مسئله جای‌گذاری ماشین‌های مجازی تاثیر به‌سزایی در کارایی مراکز داده ابری به واسطه کاهش مصرف انرژی دارد. در این تحقیق از یک روش هوشمندانه برای حل مسئله جای‌گذاری ماشین‌های مجازی استفاده شده است. مسئله جای‌گذاری به شکل یک مسئله کمینه تعریف شده و به دنبال کاهش تعداد ماشین‌های فیزیکی فعال و همچنین کاهش مصرف انرژی در

مراکز داده ابری می‌باشد. در الگوریتم ارائه شده از تکنیک‌های باینری و گسسته نیز استفاده شده است. الگوریتم ارائه شده تعداد ماشین‌های فیزیکی را به حداقل می‌رساند. این روش هم بر روی ماشین‌های فیزیکی همگن و غیر همگن اجرا شده است. نتایج حاصله نشان دهنده عملکرد صحیح الگوریتم ارائه شده می‌باشد [۲۵].

فاطمیما و همکاران در سال ۲۰۱۹ برای بر طرف کردن چالش‌های موجود در مسئله جای‌گذاری ماشین‌های مجازی در مراکز داده ابری یک الگوریتم فرا ابتکاری جدیدی بر پایه الگوریتم بهینه‌سازی گرگ خاکستری ارائه کرده‌اند. با توجه به گسترش تعداد درخواست‌های کاربران، تعداد مراکز داده ابری نیز افزایش می‌یابد. مرکز داده ابری محلی می‌باشد که ماشین‌های فیزیکی که وظیفه میزبانی از ماشین‌های مجازی را دارند، در آن قرار می‌گیرند. میلیون‌ها ماشین مجازی به وجود می‌آید که تعداد ماشین‌های فیزیکی را کاهش دهد. با توجه به این گسترش، نیاز به یک روش هوشمندانه برای مدیریت مرکز داده ابری می‌باشد. تکنیک جای‌گذاری ماشین‌های مجازی یکی از مهم‌ترین چالش‌ها در رایانش ابری می‌باشد. در این تحقیق یک روش چند هدفه بر اساس الگوریتم بهینه‌سازی گرگ خاکستری ارائه شده است. الگوریتم ارائه شده سعی در کاهش مصرف انرژی و افزایش کارایی مراکز داده ابری شده است. نتایج حاصله در مقایسه با نتایج الگوریتم‌های دیگر باعث بهبود این اهداف شده است [۲۶]. وی و همکاران در سال ۲۰۱۹ الگوریتم فرا ابتکاری نوینی بر پایه الگوریتم بهینه‌سازی کلونی مورچگان برای حل مسئله جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری ارائه داده‌اند. تکنیک مجازی‌سازی یکی از مناسب‌ترین روش‌ها در رایانش ابری بوده و با جای‌گذاری موثر ماشین‌های مجازی باعث بهبود مصرف منابع در مرکز

داده ابری شده و همچنین میزان هزینه‌ها را نیز کاهش می‌دهد. اهداف اصلی روش ارائه شده کاهش مصرف انرژی و همچنین کاهش لینک‌های ارتباطی در مراکز داده ابری می‌باشد. برای دستیابی به این اهداف، الگوریتم کلونی مورچه بهبود یافته مورد استفاده قرار گرفته است تا توازن در هزینه و مصرف انرژی به وجود بیاورد. نتایج به دست آمده از الگوریتم ارائه شده در مقایسه با روش‌های دیگر به این نکته اشاره می‌کند که الگوریتم ارائه شده باعث کاهش میزان مصرف انرژی، کاهش هزینه ارتباطات و همچنین باعث کاهش زمان اجرا در مراکز داده ابری شده است [۲۷].

لی و همکاران در سال ۲۰۱۹ یک الگوریتم فرا ابتکاری جدید بر پایه الگوریتم واکنش شیمیایی برای جای‌گذاری ماشین‌های مجازی در مراکز داده ابری ارائه داده‌اند. با گسترش روزافزون تکنیک‌های مجازی‌سازی، مراکز داده ابری با تعداد درخواست‌های بیشتر ماشین‌های مجازی مواجه می‌باشند. انتخاب یک روش جای‌گذاری بهینه برای ماشین‌های مجازی در مراکز داده ابری یکی از مهم‌ترین چالش‌ها در رایانش ابری می‌باشد. مصرف بهینه منابع و همچنین تعداد ماشین‌های فیزیکی فعال در همه مراکز داده ابری به وسیله پردازش ماشین‌های مجازی بر روی ماشین‌های فیزیکی قابل تشخیص می‌باشد. در این تحقیق الگوریتم جدیدی برای حل مسئله جای‌گذاری ارائه شده است. به حداقل رساندن مصرف انرژی و به حداکثر رساندن استفاده از منابع مهم‌ترین اهداف این تحقیق بوده که با الگوریتم ارائه شده بررسی شده است. الگوریتم فرا ابتکاری با چندین الگوریتم قبلی مقایسه شده و دارای نتایج بهتری می‌باشد [۲۸].

۳- مفاهیم پایه

در این قسمت به بیان مفاهیم پایه از جمله الگوریتم فرا ابتکاری چند نظمی و توابع آشوبناک پرداخته می‌شود.

۳-۱- الگوریتم فرا ابتکاری جمعیت سالپ

یکی از جدیدترین الگوریتم‌های بهینه‌سازی، الگوریتم بهینه‌سازی جمعیت سالپ می‌باشد. این الگوریتم به صورت یک جمعیت بوده و برای حل مسایل بهینه‌سازی به خصوص حل مسایل مهندسی ارائه شده است. ایده اصلی این الگوریتم بر اساس رفتار سالپ‌ها به صورت گروهی در داخل اقیانوس می‌باشد [۲۹]. نحوه حرکت گروهی سالپ‌ها زنجیر مانند می‌باشند. برای به دست آوردن مدل ریاضی الگوریتم جمعیت سالپ‌ها، در ابتدا جمعیت به دو گروه پیشوا و پیروی کننده‌ها تقسیم می‌شود. پیشوا سالپی می‌باشد که جلوتر از همه سالپ‌ها و در جلوی گروه یعنی اول زنجیر قرار می‌گیرد. ما بقیه اعضای گروه دنبال کننده رهبر زنجیره خود می‌باشند. با توجه به رفتار سالپ‌ها، سالپ رهبر هدایت کننده ما بقیه سالپ‌ها در حرکت می‌باشد. از آنجایی که سالپ‌ها پشت سر هم قرار گرفته‌اند هر سالپ وظیفه هدایت سالپ ما قبل خود را در زنجیره دارد.

همانند تکنیک‌های دیگر بر اساس جمعیت اولیه، وضعیت سالپ‌ها در فضای جستجو Ω بعدی بوده در حالیکه Ω تعداد متغیرهای موجود در این الگوریتم می‌باشد. نحوه قرار گرفتن سالپ‌ها در فضای جستجو در رابطه (۱) نشان داده شده است.

$$X_j^1 = \begin{cases} F_j + C_1((Ub_j - lb_j)C_2 + lb_j) & C_3 \geq 0 \\ F_j - C_1((Ub_j - lb_j)C_2 + lb_j) & C_3 < 0 \end{cases} \quad (1)$$

در رابطه (۱)، X_j^i نشان دهنده موقعیت سالپ پیشوا می‌باشد. F_j موقعیت غذا را در j ام بعد نشان می‌دهد. Ub_j و lb_j به ترتیب بالاترین و پایین‌ترین حد را در j ام بعد نشان می‌دهند. C_1 ، C_2 و C_3 اعداد تصادفی می‌باشند. با توجه به رابطه (۱) به این نتیجه می‌رسیم که سالپ پیشوا فقط به طرف غذا موقعیت خود را تغییر می‌دهد. C_1 در الگوریتم بهینه‌سازی جمعیت سالپ مهم‌ترین پارامتر تصادفی می‌باشد چرا که توازن ما بین اکتشاف و جستجو را در فضای جستجو بر قرار می‌کند. رابطه (۲) نحوه به دست آمدن مقدار C_1 را نشان می‌دهد.

$$C_1 = 2e^{-i\left(\frac{41}{L}\right)^2} \quad (2)$$

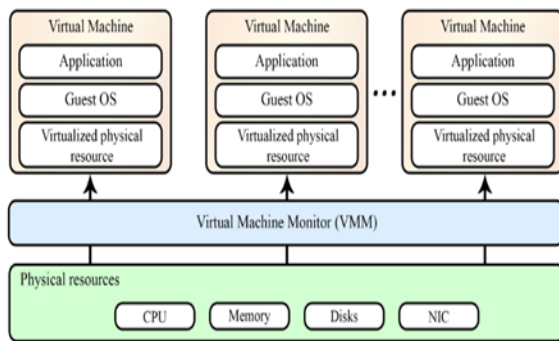
۳-۲- الگوریتم تپه‌نوردی

تپه‌نوردی یک تکنیک بهینه‌سازی متعلق به خانواده الگوریتم‌های جستجوی محلی است. یک تکنیک تکرار شونده که با یک راه حل دلخواه شروع به کار کرده و سپس تلاش می‌کند تا با تغییر بر روی یک عنصر از راه حل، به پاسخ بهتری دست پیدا کند. اگر این تغییر منجر به ایجاد یک راه حل بهتر شود، تغییر دیگری بر روی این راه حل جدید انجام خواهد گرفت. این روال تا زمانی که بهبود بیشتری در راه حل میسر نباشد ادامه می‌یابد. تپه‌نوردی اغلب در حالتی که زمان اجرای جستجو محدود است، نتیجه بهتری از سایر الگوریتم‌های هم‌نوعش تولید می‌کند. تپه‌نوردی یک الگوریتم همه‌زمانی است. حتی اگر پیش از رسیدن به پایان نیز قطع شود یک جواب قابل قبول تحویل می‌دهد.

نحوه عملکرد این الگوریتم بدین شکل می‌باشد که در ابتدا یکی از اعضای مجموعه‌ی S را به صورت تصادفی انتخاب می‌کنیم و آن را A می‌نامیم. سپس در میان اعضای S

۴- مسئله جای گذاری ماشین‌های مجازی

قرار گرفتن بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری یکی از مهم‌ترین مسایل موجود در رایانش ابری می‌باشد. جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی جز مسایل پیچیده می‌باشد. هدف از جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده در بستر رایانش ابری، به حداقل رساندن بعضی از فاکتورها مانند مصرف انرژی، جلوگیری از اتلاف منابع و غیره و همچنین به حداکثر رساندن بعضی از فاکتورها مانند کارایی بالا می‌باشد [۳۱].



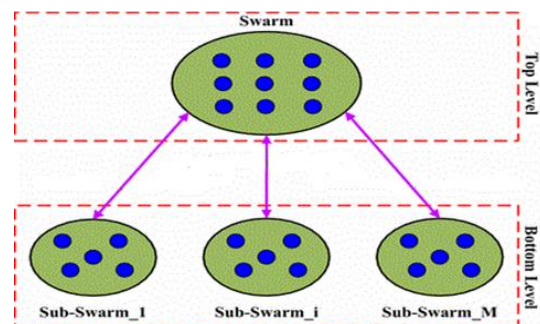
شکل ۲- مکانیزم مجازی سازی

اولین پارامتر مورد بررسی، مصرف انرژی توسط ماشین‌های فیزیکی در مراکز داده ابری می‌باشد. در تحقیقات اخیر نشان داده شده است که مصرف انرژی توسط سرورهای مراکز داده می‌تواند به صورت دقیق به وسیله یک رابطه خطی مابین مصرف انرژی و مصرف پردازنده اندازه‌گیری شود. رابطه زیر مصرف انرژی را در مراکز داده رایانش ابری نشان می‌دهد [۳۲].

که به A نزدیکند به دنبال پاسخ مناسب‌تری برای مسئله می‌گردیم. به بیانی دیگر تلاش می‌کنیم در میان اعضای که به A نزدیکند عضوی بیابیم که مقدار تابع f برای آن بیشتر از $f(A)$ باشد. سپس این روند را ادامه می‌دهیم تا به یک پیشینه نسبی برای f دست یابیم [۳۰].

۳-۳- مسئله چند جمعیتی

در مسئله چند جمعیتی به جای استفاده از یک جمعیت استاندارد، از چندین زیر جمعیت استفاده می‌شود. راهکار کلی در الگوریتم‌های بهینه‌سازی چند جمعیتی به این گونه می‌باشد که هر جمعیت بر روی یک منطقه خاص تمرکز می‌کند و فعالیت این جمعیت‌ها در ناحیه مخصوص خود به صورت مستقل می‌باشد. از آنجایی که هر جمعیت در ناحیه مستقل خود فعالیت می‌کند، بنابراین دارای اکتشاف و حرکت و بهره‌برداری مستقل از ناحیه‌های دیگر می‌باشد. با توجه به مستقل بودن جمعیت‌ها، از الگوریتم‌های اولیه متفاوتی برای جستجو استفاده می‌کنند. با توجه به انجام جستجوها در داخل جمعیت‌ها، احتمال به دست آوردن بهینه مشترک نسبت به الگوریتم‌های دیگر بهینه بسیار کاهش می‌یابد. شکل شماره (۱) نشان دهنده یک ساختار ساده از مسئله چند جمعیتی می‌باشد.



شکل ۱- ساختار مسئله چند جمعیتی

$$R_p^{wastage} = \frac{|NR_p^{cpu} - NR_p^{mem}| + \epsilon}{U_p^{cpu} + U_p^{mem}} \quad (6)$$

در رابطه (۶)، $R_p^{wastage}$ نشان دهنده اتلاف منابع ماشین‌های فیزیکی در مرکز داده ابری می‌باشد. NR_p^{cpu} متغیری می‌باشد که میزان قدرت باقی مانده CPU ماشین فیزیکی P را در مرکز داده ابری نشان می‌دهد. NR_p^{mem} این متغیر نشان دهنده باقی مانده حافظه در ماشین فیزیکی P می‌باشد. U_p^{cpu} ، میزان پردازنده استفاده شده در ماشین فیزیکی P را بر حسب میلیون دستورات‌عمل در هر ثانیه نشان می‌دهد. میزان حافظه مورد استفاده به وسیله ماشین‌های مجازی بر روی ماشین فیزیکی P به وسیله متغیر U_p^{mem} نشان داده می‌شود.

جای‌گذاری بهینه ماشین‌های مجازی باعث کاهش ترافیک در مراکز داده ابری شده و هر مقدار ترافیک کمتر شود، سرعت انتقال و در نتیجه کارایی در مراکز داده ابری بیشتر می‌شود. بنابراین هدف به حداقل رساندن ترافیک ما بین گره‌های موجود در مراکز داده ابری می‌باشد. در مراکز داده دارای دو نوع ترافیک می‌باشد، ترافیک ما بین ماشین‌های مجازی و ترافیک ما بین ماشین‌های مجازی با ماشین‌های فیزیکی. میزان ترافیک از طریق رابطه (۷) نمایش داده می‌شود.

$$T_{cost} = \sum_{i,j=1}^n a_{ij} b_{hp} * \pi(j) + \sum_{i=1}^n e_i g_i \pi(i) \quad (7)$$

در رابطه (۷) قسمت اول نشان دهنده ترافیک ما بین ماشین‌های مجازی و قسمت دوم نشان دهنده ترافیک ماشین‌های مجازی با ماشین‌های فیزیکی می‌باشد. در رابطه (۷)، a_{ij} نشان دهنده ترافیک ما بین ماشین مجازی i و ماشین مجازی j می‌باشد. b_{hp} ، متغیری برای نشان دادن هزینه ارتباط ما بین ماشین فیزیکی h با ماشین فیزیکی P می‌باشد.

$$P_p^{power} = (P_p^{busy} - P_p^{idle}) * U_p^{cpu} + P_p^{idle} \quad (4)$$

در رابطه (۴)، P_p^{power} نشان دهنده مصرف انرژی ماشین فیزیکی p می‌باشد. P_p^{busy} ، میزان مصرف انرژی ماشین فیزیکی p را در زمان فعال بودن نشان می‌دهد. P_p^{idle} ، میزان مصرف انرژی را نشان می‌دهد که در آن زمان ماشین فیزیکی p بیکار می‌باشد. U_p^{cpu} ، نشان دهنده میزان استفاده ماشین فیزیکی p از پردازنده به صورت میلیون دستورات‌عمل در هر ثانیه می‌باشد. با توجه به رابطه (4) میزان مصرف انرژی با میزان مصرف پردازنده یک رابطه خطی دارد. بنابراین به هر اندازه میزان مصرف پردازنده ماشین‌های فیزیکی بیشتر باشد، میزان انرژی مصرفی در مرکز داده نیز افزایش می‌یابد. بنابراین کل میزان انرژی مصرفی مرکز داده ابری به صورت زیر محاسبه می‌گردد.

$$\sum_{p=1}^m P_p^{power} = \sum_{p=1}^m b_p * (P_p^{busy} - P_p^{idle}) * \left(\sum_{v=1}^n (a_{vp} \cdot C_v) + P_p^{idle} \right) \quad (5)$$

در رابطه (۵)، a و b متغیرهای باینری بوده که نشان دهنده فعال و غیر فعال بودن ماشین‌های فیزیکی در مرکز داده ابری می‌باشد. متغیر C_v میزان استفاده از پردازنده ماشین فیزیکی میزبان به وسیله ماشین مجازی را نشان می‌دهد.

یکی دیگر از معیارهای مهم در مساله جای‌گذاری ماشین‌های مجازی، جلوگیری از اتلاف منابع در مراکز داده ابری می‌باشد. هر سرور موجود در مرکز داده دارای یک منبع سخت‌افزاری بوده و ماشین‌های مجازی مختلفی را می‌تواند میزبانی کند. منابع استفاده نشده بر روی سرورها باید به صورت مناسب مدیریت شود. رابطه (۶) مقدار اتلاف منابع را نشان می‌دهد.

در مسئله جای‌گذاری بهینه محدودیت‌های وجود دارد که این محدودیت‌ها عبارتند از:

$$\sum_{p=1}^m a_{vp} = 1 \quad (13)$$

$$\sum_{v=1}^n a_{vp} \cdot C_v \leq T_p^{cpu} \cdot b_p \quad (14)$$

$$\sum_{v=1}^n a_{vp} \cdot M_v \leq T_p^{mem} \cdot b_p \quad (15)$$

رابطه شماره (۱۳) نشان دهنده این می‌باشد که هر ماشین مجازی اجازه دارد که فقط بر روی یک ماشین فیزیکی در مرکز داده ابری قرار بگیرد. روابط (۱۴) و (۱۵) به این نکته اشاره می‌کنند که مجموع پردازنده و حافظه مورد نیاز ماشین‌های مجازی قرار گرفته بر روی یک ماشین فیزیکی، بیشتر از ظرفیت پردازنده و حافظه آن ماشین فیزیکی نباشد.

۵- الگوریتم پیشنهادی

یک مرکز داده ابری با چندین نوع ماشین فیزیکی با سخت افزارهای متفاوت $D_{end-end}$ و چندین نوع ماشین مجازی مختلف با نیازمندی‌های متفاوت وجود دارد. ماشین فیزیکی دارای یکسری منابع ای می‌باشد که اگر جای‌گذاری ماشین‌های مجازی به نحوی باشد که از همه منابع ماشین فیزیکی به صورت کامل استفاده شود، آنگاه ماشین فیزیکی میزبان از تمامی توان پردازشی خود استفاده کرده و منبع‌ای هدر نمی‌رود. در این حالت یک ماشین فیزیکی دیگر وظیفه میزبانی را برای ماشین‌های مجازی بر عهده می‌گیرد. اگر جای‌گذاری ماشین‌های مجازی به صورت بهینه انجام نشود، منابع سخت‌افزاری ماشین‌های فیزیکی در مرکز داده ابری هدر می‌رود. با جای‌گذاری بهینه ماشین‌های مجازی می‌توان از توان پردازشی ماشین‌های فیزیکی به صورت کامل استفاده نمود که باعث جلوگیری از هدر رفتن منابع سخت‌افزاری

e_i نشان دهنده ترافیک ما بین ماشین مجازی i با یک گره خارج از مرکز داده ابری می‌باشد. متغیر $g_{\pi(i)}$ هزینه ارتباط ماشین مجازی i بر روی ماشین فیزیکی P با سوئیچ می‌باشد.

یکی دیگر از مهم‌ترین پارامترها نقش زیادی در کارایی مراکز داده ابری دارد، میزان تاخیر می‌باشد. تاخیر ارسال نشان دهنده مدت زمانی می‌باشد که بسته از مبدا تا مقصد طول می‌کشد، که برسد. میزان تاخیر انتها به انتها با استفاده از رابطه زیر محاسبه می‌شود.

$$D_{end-end} = N(D_{trans} + D_{prop} + D_{proc} + D_{queue}) \quad (8)$$

نشان دهنده میزان تاخیر ارسال داده از مبدا به مقصد می‌باشد. N تعداد بسته‌های انتقالی را نشان می‌دهد. D_{trans} نشان دهنده مدت زمانی می‌باشد که نیاز است تا

همه بیت‌های بسته بر روی کانال ارتباطی ارسال شوند. D_{prop} ، مدت زمانی را شامل می‌شود که بیت‌های بسته از مبدا به مقصد برسد. D_{proc} ، مدت زمان لازم برای پردازش بسته سرایند به وسیله مسیریاب‌ها را شامل می‌شود. D_{queue} ، مدت زمانی را شامل می‌شود که بسته برای اجرا شدن در صف منتظر می‌ماند.

هدف از جای‌گذاری بهینه به حداقل رساندن معیارهای گفته شده می‌باشد. بنابراین توابع هدف به صورت زیر تعریف می‌شوند.

$$\text{Minimize } \sum_{p=1}^m P_p^{power} \quad (9)$$

$$\text{Minimize } \sum_{p=1}^m R_p^{wastage} \quad (10)$$

$$\text{Minimize } T_{cost} \quad (11)$$

$$\text{Minimize } D_{end-end} \quad (12)$$

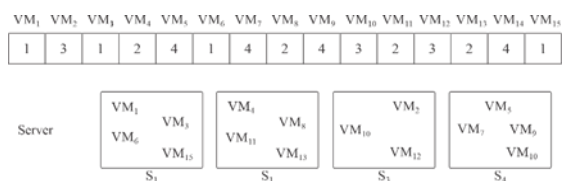
در مرکز داده ابری می‌شود. برای داشتن یک مرکز داده ابری هوشمند، باید مصرف منابع در ماشین‌های فیزیکی به حداقل برسد.

در این مقاله از ترکیب الگوریتم فرا ابتکاری جمعیت سالپ چند جمعیتی گسسته و الگوریتم تپه‌نوردی برای حل مسئله جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری استفاده شده است. هدف استفاده از این راهکار به حداقل رساندن میزان مصرف انرژی و هدر رفتن منابع در مرکز داده ابری می‌باشد.

جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی یک مسئله گسسته می‌باشد. نحوه عملکرد الگوریتم‌های فرا ابتکاری به صورت پیوسته انجام می‌شود. بنابراین برای انجام جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مرکز داده ابری از عملگرهای جدیدی استفاده می‌کنیم که مسئله جای‌گذاری را به صورت گسسته با الگوریتم پیشنهادی حل نماییم.

در روش پیشنهادی، عملگرهای جدید شامل عملگرهای ضرب، منهای و جمع می‌باشد. در الگوریتم پیشنهادی از یک عملگر جدید منهای، به جای جمع از عملگر برش و به جای عملگر ضرب از عملگر تعویض استفاده خواهیم کرد.

شکل (۳) نحوه قرار گرفتن ماشین‌های مجازی را بر روی ماشین‌های فیزیکی نشان می‌دهد.



شکل ۳- نگاشت ماشین‌های مجازی

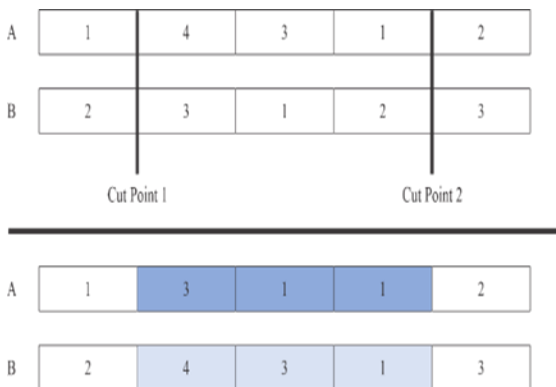
در الگوریتم پیشنهادی از یک عملگر منهای جدید استفاده شده است. دو ذره A و ذره B به صورت زیر در نظر گرفته شده است.

A	1	2	3	1	2
B	2	2	1	2	3

شکل ۴- نمونه‌ای از ذرات

در عملگر منهای جدید، اگر تابع برازندگی ذره A از تابع برازندگی ذره B بیشتر باشد، ذره A را به عنوان جواب انتخاب می‌کند در غیر این صورت ذره B که دارای تابع برازندگی بیشتری می‌باشد، به عنوان جواب عملگر منهای در الگوریتم پیشنهادی انتخاب می‌شود.

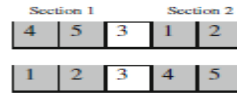
در الگوریتم پیشنهادی به جای عملگر جمع، از عملگر برش استفاده شده است. در این عملگر به صورت تصادفی دو نقطه از ذرات انتخاب شده و از آن نقاط برش انجام شده و محتوای ما بین این دو برش در دو ذره عوض می‌شوند. جواب نهایی به صورت تصادفی یکی از ذرات انتخاب خواهد شد.



شکل ۵- عملگر جمع جدید

در الگوریتم پیشنهادی به جای عملگر ضرب، از عملگر تعویض استفاده شده است. در این روش دو زیر رشته به صورت تصادفی از ذره انتخاب شده و محل آنها در ذره

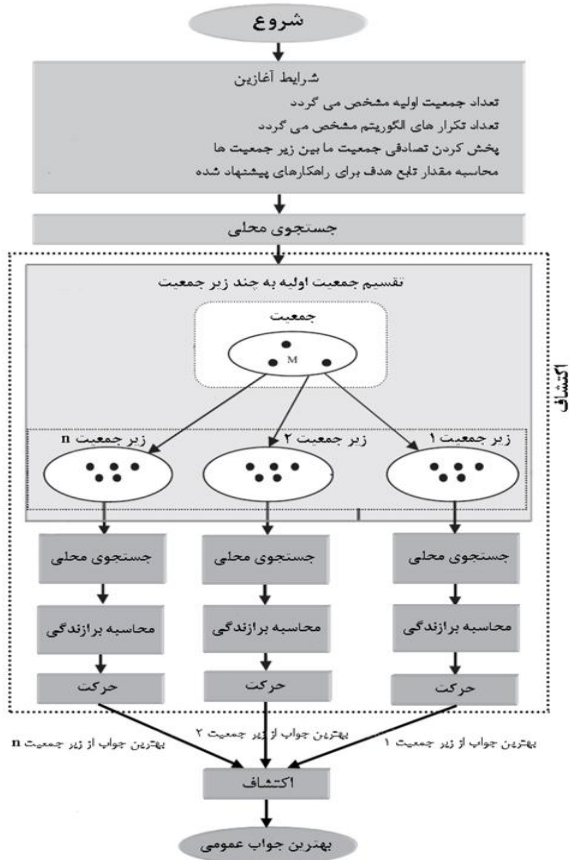
تعویض می‌گردد. شکل (۶) نحوه عملکرد عملگر ضرب جدید نشان داده شده است.



شکل ۶ - عملگر جدید ضرب

در این مقاله از روش چند جمعیتی ارباب - برده استفاده شده است. به جای استفاده از یک جمعیت استاندارد، از چند زیر جمعیت استفاده شده است. جمعیت اولیه به هنگام شروع اجرا به صورت تصادفی به چندین زیر جمعیت تقسیم می‌شود. هر کدام از این زیر جمعیت‌ها به صورت مستقل می‌باشند.

الگوریتم پیشنهادی بر روی هر کدام از این زیر جمعیت‌ها اجرا می‌شود. بهترین جواب بعد از اجرای الگوریتم پیشنهادی بر روی زیر جمعیت‌ها انتخاب شده و از نتایج به دست آمده از زیر جمعیت‌ها، بهترین جواب انتخاب می‌شود. نحوه کار در شکل (۷) نشان داده شده است.



شکل ۷- نحوه اجرای الگوریتم پیشنهادی بر روی ساختار چند جمعیتی

الگوریتم پیشنهادی کار خود را به وسیله بررسی اولیه جمعیت موجود با استفاده از الگوریتم فرا ابتکاری چندنظمی گسسته چند هدفه و توابع آشوبناک شروع می‌کند. هر ذره ارزیابی شده و تابع برازندگی آن بر اساس معیارهای موجود در روابط (۴) و (۶) تعیین می‌شود. بر اساس سخت افزاری که هر ماشین مجازی دارد، بر روی یک ماشین فیزیکی قرار می‌گیرد. اگر ماشین‌های فیزیکی میزبان توانایی تخصیص منبع لازم را برای ماشین‌های مجازی نداشته باشند، آنگاه ماشین‌های مجازی از یک میزبان به میزبان دیگر مهاجرت می‌کنند. الگوریتم پیشنهادی چندین بار تکرار شده تا به یک نتیجه مطلوب از اجرا برسد. روند کار الگوریتم پیشنهادی در شکل شماره (۸) نشان داده شده است.

۶- نتایج شبیه سازی

در این بخش به بررسی نتایج به دست آمده، از حل مساله جای گذاری بهینه ماشین های مجازی بر روی ماشین های فیزیکی پرداخته می شود. هدف از جای گذاری بهینه ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده در بستر رایانش ابری، به حداقل رساندن بعضی از فاکتورها مانند مصرف انرژی، جلوگیری از اتلاف منابع و غیره و همچنین به حداکثر رساندن بعضی از فاکتورها مانند کارایی بالا می باشد. در این مقاله، هدف به حداقل رساندن مصرف انرژی، اتلاف منابع، پهنای باند و تاخیر ارسال می باشد.

در این مقاله، از دو معماری VL2 و Three Tier در ساختار ابری برای جای گذاری ماشین های مجازی بر روی ماشین های فیزیکی با استفاده از الگوریتم پیشنهادی استفاده شده است.

در این مقاله برای شبیه سازی مسئله جای گذاری ماشین های مجازی بر روی ماشین های فیزیکی از زبان برنامه نویسی متلب استفاده شده است. در این فضای شبیه سازی از محیط ابری Amazon EC2 استفاده شده است. در فضای شبیه سازی ۲۵۰ سرور ناهمگن استفاده شده است. هر سرور دارای یک سری منابع فیزیکی بوده که مشخصات آن در جدول شماره (۱) نشان داده شده است.

جدول ۱- منابع سخت افزاری ماشین های فیزیکی

نام ماشین فیزیکی	پردازنده (MIPS)	حافظه (GB)	فضای ذخیره سازی (GB)	پهنای باند (MB)
HP ProLiant G4	۵۱۲	۱۰۲۴	۴	۳۷۲۰
HP ProLiant G5	۱۰۲۴	۲۰۴۸	۴	۵۳۲۰

```

Input: PM list, VM list
Output: VM mapping
//Setting Parameters
Set nSol, max iteration, X0 and G0
//initialize all solutions

For i=1 : nSol
    Initialize Sol(i) randomly
    Calculate cost of Sol(i)
End for
//find thr best solution
[~,indx] = min([Sol.cost]);
Gbest = Sol(index);
//main loop
For iteration = 1: max iteration
    For j=1:nSol
        Update Sol(i) using proposed SSA algorithm
        Calculate cost of newSol(i)
        // applying hill climbing approach
        If mod(iteration)==0
            If cost of newSol(i) < cost of Sol(i)
                Sol(i)=newSol(i);
            Else
                Sol(i)=Gbest;
            Endif
        Else
            Sol(i)=newSol(i);
        Endif
    End for
    Update Gbest
End for
Return Gbest as output
    
```

شکل ۸- روند کار الگوریتم پیشنهادی

ما از تعداد ، ۲۰۰ ، ۴۰۰ و ۶۰۰ ماشین مجازی برای جای گذاری در مرکز داده استفاده کرده ایم. هر ماشین مجازی نیازمند منابع سخت افزاری می باشد که توسط ماشین فیزیکی که بر روی آن قرار می گیرد، باید تامین گردد.

جدول ۲- منابع سخت افزاری ماشین های مجازی

پهنای باند (MB)	فضای ذخیره سازی (GB)	حافظه (GB)	پردازنده (MIPS)	نام ماشین مجازی
۲۵۶	۴۰	۶۱۳	۵۰۰	Micro
۵۱۲	۱۲۰	۱۷۰۰	۱۰۰۰	Small
۷۶۸	۱۶۰	۳۴۰۰	۲۰۰۰	Extra Large
۲۵۶	۸۰	۸۵۰	۲۵۰۰	High-CPU medium

ماشین های مجازی با نیازمندی های متفاوت سخت افزاری بر روی ماشین های فیزیکی در مرکز داده ابری قرار می گیرند. ماشین های فیزیکی در مرکز داده ابری سخت افزار مورد نیاز را در اختیار ماشین های مجازی قرار داده و اگر منابع سخت افزاری ماشین های مجازی به وسیله ماشین های فیزیکی میزبان تامین نگردد و یا نیاز به سخت افزار بیشتری در هنگام پردازش نیاز پیدا کنند و این نیاز به وسیله ماشین فیزیکی میزبان تامین نگردد، ماشین مجازی از روی ماشین فیزیکی میزبان به یک ماشین فیزیکی در مرکز داده ابری مهاجرت می کند. هر چقدر تعداد مهاجرت ماشین های مجازی در مرکز داده ابری کمتر باشد، تعداد ماشین های فعال کاهش یافته و در نتیجه کارایی مرکز داده افزایش می یابد. نتایج به دست آمده از الگوریتم پیشنهادی با نتایج الگوریتم های کار شده در این زمینه از جمله GSA، FCFS، FF، GA، ACO مقایسه خواهد شد. جداول (۳)، (۴) و (۵) نتایج جای گذاری ماشین های مجازی را بر روی ماشین های فیزیکی با استفاده از معماری VL2 را نشان می دهند.

جدول ۳: نتایج جای گذاری ۲۰۰ ماشین مجازی در ماشین فیزیکی VL2 در معماری

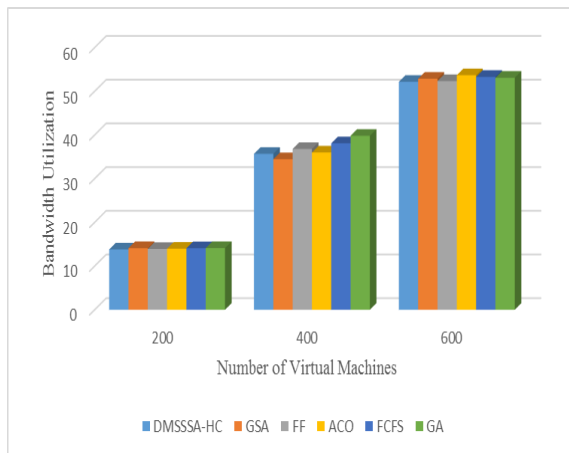
تاخیر ارسال (%)	پهنای باند (%)	مدیریت منابع	مصرف انرژی (W)	الگوریتم
۹.۴۳	۱۳.۸	۵۷۲.۸۵۷۷	۱.۵۶*۱۰ ^۶	DMSSSA-HC
۱۰.۰۶	۱۴.۱	۵۸۱.۷۳۱۹	۱.۶۲*۱۰ ^۶	GSA
۱۰.۴۲	۱۳.۹۲	۵۹۰.۰۹۲۳	۱.۵۹*۱۰ ^۶	FF
۱۰.۰۲	۱۳.۹۸	۵۷۹.۶۵۲۱	۱.۷۳*۱۰ ^۶	ACO
۱۰.۰۹	۱۴.۰۸	۵۸۴.۴۴۱۹	۱.۶۸*۱۰ ^۶	FCFS
۹.۶۵	۱۴.۱۱	۵۸۲.۲۳۴۲	۱.۸۲*۱۰ ^۶	GA

جدول ۴: نتایج جای گذاری ۴۰۰ ماشین مجازی در ماشین فیزیکی VL2 در معماری

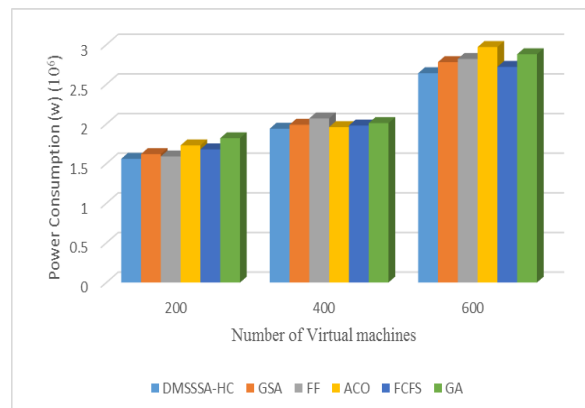
تاخیر ارسال (%)	پهنای باند (%)	مدیریت منابع	مصرف انرژی (W)	الگوریتم
۲۵.۱۸	۳۵.۶۷	۶۲۳.۴۳۲۹	۱.۹۴*۱۰ ^۶	DMSSSA-HC
۲۵.۹۸	۳۴.۴۵	۶۲۹.۱۳۳۹	۱.۹۹*۱۰ ^۶	GSA
۲۶.۲۴	۳۶.۷۹	۶۴۷.۶۱۳۹	۲.۰۷*۱۰ ^۶	FF
۲۵.۶۷	۳۵.۹۸	۶۲۸.۵۶۱۲	۱.۹۶*۱۰ ^۶	ACO
۲۶.۷۱	۳۸.۰۹	۶۲۴.۹۱۴۱	۱.۹۸*۱۰ ^۶	FCFS
۲۶.۹۱	۳۹.۸۱	۶۳۰.۵۰۶۹	۲.۰۱۱*۱۰ ^۶	GA

جدول ۵: نتایج جای گذاری ۶۰۰ ماشین مجازی در ماشین فیزیکی VL2 در معماری

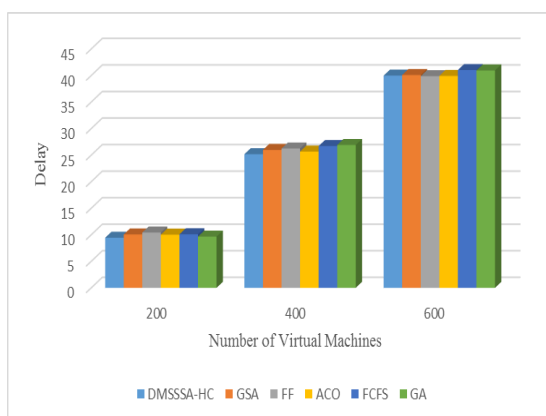
تاخیر ارسال (%)	پهنای باند (%)	مدیریت منابع	مصرف انرژی (W)	الگوریتم
۳۹.۹۸	۵۲.۱۷	۶۸۲.۴۳۲۹	۲.۶۴*۱۰ ^۶	DMSSSA-HC
۴۰.۰۹	۵۲.۸۷	۶۸۴.۱۳۳۹	۲.۷۸*۱۰ ^۶	GSA
۳۹.۸۷	۵۲.۳۳	۶۸۳.۶۱۳۹	۲.۸۲*۱۰ ^۶	FF
۳۹.۹۲	۵۳.۶۷	۶۸۹.۵۶۱۲	۲.۹۷*۱۰ ^۶	ACO
۴۱.۰۳	۵۳.۲۹	۶۹۲.۹۱۴۱	۲.۷۲*۱۰ ^۶	FCFS
۴۰.۹۵	۵۳.۰۹	۶۹۰.۵۰۶۹	۲.۸۸*۱۰ ^۶	GA



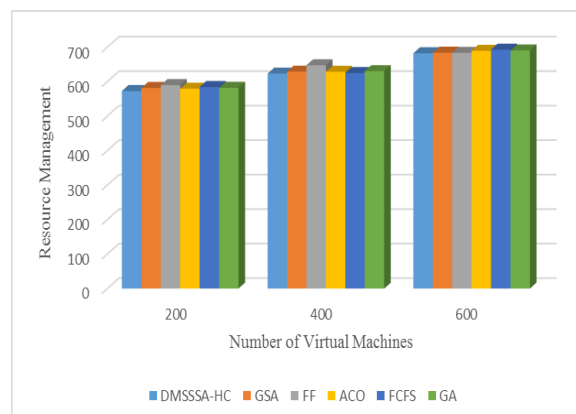
شکل ۱۰: نمودار مصرف پهنای باند برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین VL2 مجازی با معماری



شکل ۸: نمودار مصرف انرژی برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین VL2 مجازی با معماری



شکل ۱۱: نمودار تاخیر ارسال برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین VL2 مجازی با معماری

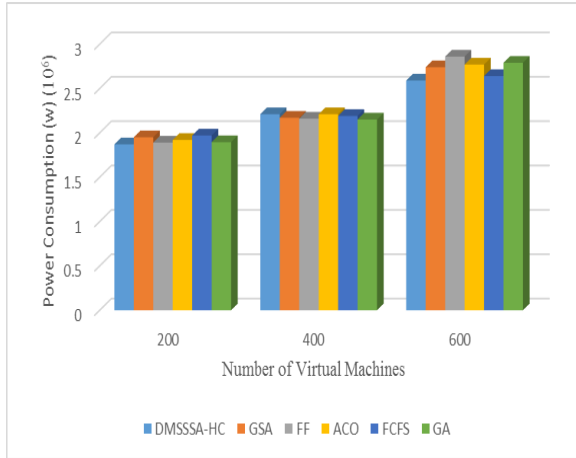


شکل ۹: نمودار مدیریت منابع برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین VL2 مجازی با معماری

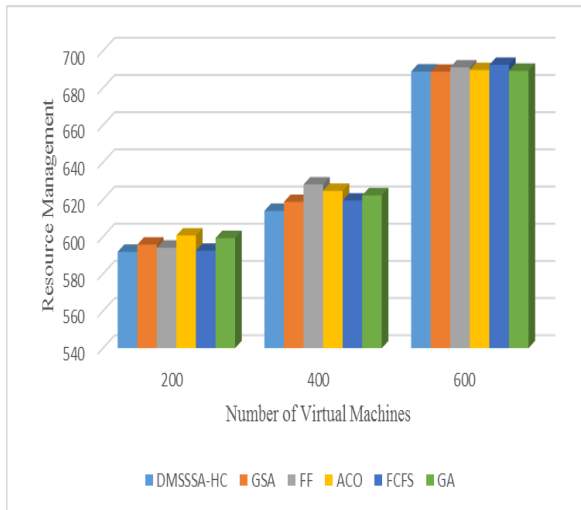
جداول زیر نتایج جای گذاری ماشین های مجازی را بر روی ماشین های فیزیکی با استفاده از معماری Three Tier را نشان می دهند.

جدول ۶: نتایج جای گذاری ۲۰۰ ماشین مجازی در ماشین فیزیکی Three Tier در معماری

الگوریتم	مصرف انرژی (W)	مدیریت منابع	پهنای باند (%)	تأخیر ارسال (%)
DMSSSA-HC	1.87*10 ⁶	591.8577	12.9	11.16



شکل ۱۲: نمودار مصرف انرژی برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین مجازی با معماری Three Tier



شکل ۱۳: نمودار مدیریت منابع برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین مجازی با معماری

GSA	۱.۹۵*۱۰ ^۶	۵۹۵.۷۳۱۹	۱۳.۹	۱۰.۸۵
FF	۱.۸۹*۱۰ ^۶	۵۹۴.۰۹۲۳	۱۳.۷۶	۱۰.۹۲
ACO	۱.۹۲*۱۰ ^۶	۶۰۰.۶۵۲۱	۱۳.۰۵	۱۱.۳۷
FCFS	۱.۹۷*۱۰ ^۶	۵۹۲.۴۴۱۹	۱۳.۴۴	۱۱.۵۶
GA	۱.۸۹۵*۱۰ ^۶	۵۹۹.۲۳۴۲	۱۳.۹۴	۱۱.۰۹

جدول ۷: نتایج جای گذاری ۴۰۰ ماشین مجازی در ماشین

Three Tier فیزیکی در معماری

الگوریتم	مصرف انرژی (W)	مدیریت منابع	پهنای باند (%)	تاخیر ارسال (%)
DMSSSA-HC	۲.۲۱*۱۰ ^۶	۶۱۳.۸۵۷۷	۳۹.۱۷	۲۸.۱۹
GSA	۲.۱۷*۱۰ ^۶	۶۱۸.۷۳۱۹	۳۹.۴۳	۲۷.۹۲
FF	۲.۱۶*۱۰ ^۶	۶۲۸.۰۹۲۳	۳۹.۸۷	۲۷.۸۷
ACO	۲.۲۱*۱۰ ^۶	۶۲۴.۶۵۲۱	۴۰.۱۵	۲۹.۸۹
FCFS	۲.۱۹*۱۰ ^۶	۶۱۹.۴۴۱۹	۳۹.۵۲	۲۸.۷۲
GA	۲.۱۵*۱۰ ^۶	۶۲۲.۲۳۴۲	۴۰.۱۹	۲۸.۵۵

جدول ۸: نتایج جای گذاری ۶۰۰ ماشین مجازی در ماشین

Three Tier فیزیکی در معماری

الگوریتم	مصرف انرژی (W)	مدیریت منابع	پهنای باند (%)	تاخیر ارسال (%)
DMSSSA-HC	۲.۵۹*۱۰ ^۶	۶۸۸.۸۵۷۷	۵۶.۱۹	۴۲.۳۳
GSA	۲.۷۴*۱۰ ^۶	۶۸۸.۷۳۱۹	۵۷.۰۹	۴۲.۴۵
FF	۲.۸۶*۱۰ ^۶	۶۹۱.۰۹۲۳	۵۶.۷۸	۴۲.۳۳
ACO	۲.۷۷*۱۰ ^۶	۶۸۹.۶۵۲۱	۵۸.۹۱	۴۲.۷۸
FCFS	۲.۶۴*۱۰ ^۶	۶۹۲.۴۴۱۹	۵۶.۷۲	۴۳.۰۳
GA	۲.۷۹*۱۰ ^۶	۶۸۹.۲۳۴۲	۵۶.۳۲	۴۲.۸۴

سخت افزاری ماشین های فیزیکی بوده، به طوری که کمترین اتلاف انرژی را داشته باشد. با توجه به نتایج حاصله، اتلاف منابع با استفاده از راهکارهای پیشنهادی نسبت به نتایج الگوریتم های مقایسه شده باعث به کاهش اتلاف منابع شده است. برای تحقق اهداف بالا استفاده حداکثری از منابع سخت افزاری ماشین های فیزیکی فعال در مرکز داده ابری باید انجام گیرد. بنابراین در مسئله جای گذاری ماشین های مجازی هر چقدر تعداد ماشین های فیزیکی فعال کاهش یابد، باعث افزایش کارایی مرکز ابری از جنبه های مختلف از جمله کاهش هزینه می شود. همچنین راهکار پیشنهادی باعث بهینه شدن و کاهش پهنای باند در مرکز داده ابری شده است. با بررسی نتایج به دست آمده، میزان تاخیر در ارسال به وسیله الگوریتم پیشنهادی کاهش یافته است. در کل می توان کارایی راهکار پیشنهادی را در موارد زیر بیان کرد:

۱- الگوریتم پیشنهادی، تعداد ماشین های فعال را در مرکز داده ابری به حداقل رسانیده است.

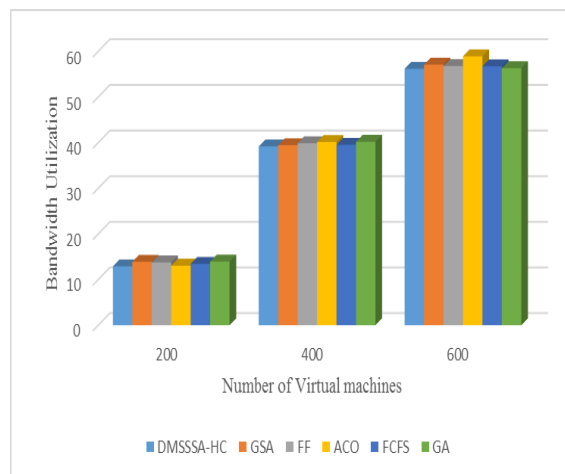
۲- الگوریتم پیشنهادی، توازن بسیار مناسبی در مصرف منابع بر روی هر ماشین فیزیکی در مرکز داده ابری به وجود آورده است.

۳- با توجه به حداقل رسیدن تعداد ماشین های فیزیکی در مرکز داده ابری، میزان مصرف انرژی نیز کاهش یافته است.

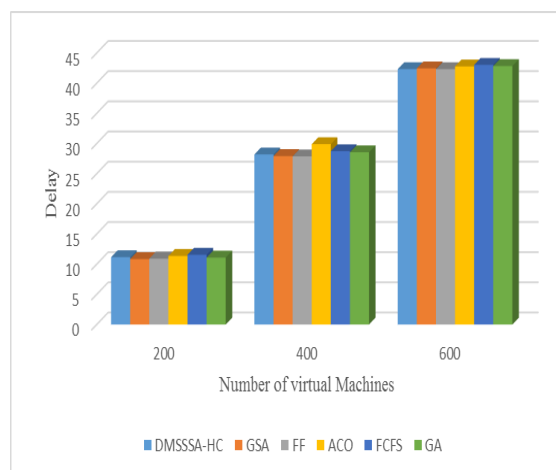
۴- الگوریتم پیشنهادی، میزان استفاده از منابع را در مرکز داده ابری بهبود داده است.

۵- الگوریتم پیشنهادی، میزان تاخیر ارسال را در مرکز داده ابری کاهش داده و باعث افزایش کارایی سیستم شده است.

۶- در الگوریتم GSA و GA توازن استفاده از منابع اصلا در نظر گرفته نشده است و همچنین هزینه مهاجرت



شکل ۱۴: نمودار مصرف پهنای باند برای ۲۰۰، ۴۰۰ و ۶۰۰ Three Tier ماشین مجازی با معماری



شکل ۱۵: نمودار تاخیر ارسال برای ۲۰۰، ۴۰۰ و ۶۰۰ ماشین Three Tier مجازی با معماری

با توجه به نتایج به دست آمده، الگوریتم پیشنهادی باعث بهبود و افزایش کارایی مرکز داده ابری شده است. با توجه به نتایج مصرف انرژی ماشین های فیزیکی فعال در مرکز داده ابری، نشان دهنده این می باشد که مصرف انرژی ماشین های فیزیکی در مرکز داده ابری با استفاده از راهکار پیشنهادی نسبت به نتایج سایر الگوریتم های پیشنهادی کاهش یافته و بهینه تر شده است. یکی دیگر از اهداف مهم در مسئله جای گذاری کاهش منابع مورد استفاده می باشد. هدف استفاده حداکثری از منابع

ماشین‌های مجازی بسیار بالا می‌باشد. در الگوریتم پیشنهادی اتلاف منابع به حداقل رسیده و هزینه مهاجرت بسیار پایین می‌باشد.

۷- نتیجه‌گیری

تخصیص ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری یکی از مهمترین چالش‌ها در رایانش ابری می‌باشد. اگر جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی به صورت بهینه نباشد، باعث افزایش فعالیت تعداد ماشین‌های فیزیکی شده و در نتیجه مصرف انرژی و اتلاف منابع را به همراه دارد. کاهش تعداد ماشین‌های فیزیکی فعال تاثیر به‌سزایی در کارایی مراکز داده ابری دارد. بنابراین یک جای‌گذاری بهینه باعث کاهش تعداد ماشین‌های مجازی شده و به مراتب مصرف انرژی و اتلاف منابع را در مراکز ابری به حداقل می‌رساند. در این مقاله، به بررسی چگونگی قرار گرفتن بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده رایانش ابری پرداخته و مسئله جای‌گذاری ماشین‌های مجازی را با استفاده از ترکیب الگوریتم فرا ابتکاری جمعیت سالپ چند جمعیتی گسسته و الگوریتم تپه‌نوردی حل شده است. در روش پیشنهادی، اولین هدف به حداقل رساندن مصرف انرژی در مرکز داده ابری به واسطه کاهش تعداد ماشین‌های فیزیکی فعال در آن می‌باشد. همچنین در روش پیشنهادی، با جای‌گذاری مناسب ماشین‌های مجازی بر روی ماشین‌های فیزیکی، استفاده بهینه از منابع سخت‌افزاری ماشین‌های فیزیکی در مراکز ابری به حداکثر می‌رسد. جلوگیری و به حداقل رساندن اتلاف منابع در مراکز داده ابری از دیگر اهداف روش پیشنهادی می‌باشد. روش پیشنهادی باعث استفاده حداکثری از منابع سخت‌افزاری ماشین‌های فیزیکی در مراکز داده ابری شده است. الگوریتم پیشنهادی تعداد مهاجرت‌های ماشین‌های فیزیکی را

در مراکز داده ابری به حداقل رسانیده و باعث افزایش کارایی شده است.

در نهایت، نتایج به دست آمده از الگوریتم پیشنهادی، با نتایج سایر الگوریتم‌های فرا ابتکاری مقایسه شده و نتایج به دست آمده نشان دهنده این است که الگوریتم پیشنهادی در بسیاری از موارد بهتر از نتایج سایر الگوریتم‌ها عمل کرده است.

در کارهای آینده سعی در کاهش SLA در مراکز داده ابری را خواهیم داشت. همچنین سعی در محاسبه تعداد مهاجرت‌های تعداد ماشین‌های مجازی و تعداد ماشین‌های فیزیکی خاموش شده در حین جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری را خواهیم داشت.

- [1] R. Bao, "Performance Evaluation for Traditional Virtual Machine Placement Algorithms in the Cloud," in International Conference on Internet of Vehicles, pp. 225-231, 2016.
- [2] M. Masdari, S. ValiKardan, Z. Shahi, and S. I. Azar, "Towards workflow scheduling in cloud computing: a comprehensive analysis," Journal of Network and Computer Applications, Vol. 66, pp. 64-82, 2016.
- [3] M. Masdari and M. Jalali, "A survey and taxonomy of DoS attacks in cloud computing," Security and Communication Networks, Vol. 9, No. 16, pp. 3724-3751, 2016.
- [4] K. Braiki and H. Youssef, "Multi-Objective Virtual Machine Placement Algorithm Based on Particle Swarm Optimization," in 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), pp. 279-284, 2018.
- [5] J. Yan, H. Zhang, H. Xu, and Z. Zhang, "Discrete PSO-based workload optimization in virtual machine placement," Personal and Ubiquitous Computing, Vol. 22, No. 3, pp. 589-596, 2018.
- [6] S. K. Addya, A. K. Turuk, B. Sahoo, M. Sarkar, and S. K. Biswash, "Simulated annealing based VM placement strategy to maximize the profit for Cloud Service Providers", Engineering science and technology, an international journal, Vol. 20, No. 4, pp. 1249-1259, 2017.
- [7] M. Masdari and A. Khoshnevis, "A survey and classification of the workload forecasting methods in cloud computing," Cluster Computing, pp. 1-26, 2019.
- [8] Y. Qin, H. Wang, F. Zhu, and L. Zhai, "A multi-objective ant colony system algorithm for virtual machine placement in traffic intense data centers," IEEE access, Vol. 6, pp. 58912-58923, 2018.
- [9] T. Shabeera, S. M. Kumar, S. M. Salam, and K. M. Krishnan, "Optimizing VM allocation and data placement for data-intensive applications in cloud using ACO metaheuristic algorithm," Engineering Science and Technology, an International Journal, Vol. 20, No. 2, pp. 616-628, 2017.
- [10] X.-F. Liu, Z.-H. Zhan, J. D. Deng, Y. Li, T. Gu, and J. Zhang, "An energy efficient ant colony system for virtual machine placement in cloud computing," IEEE Transactions on Evolutionary Computation, Vol. 22, No. 1, pp. 113-128, 2016.
- [11] M. Masdari and M. Zangakani, "Green Cloud Computing Using Proactive Virtual Machine Placement: Challenges and Issues," Journal of Grid Computing, pp. 1-33, 2019.
- [12] R. Asemi, E. Doostadigh, M. Ahmadi, and H. T. Malazi, "Energy efficiency in virtual machines allocation for cloud data centers using the imperialist competitive algorithm," in 2015 IEEE Fifth International Conference on Big Data and Cloud Computing, pp. 62-67, 2015.
- [13] F. B. Hassen, Z. Brahmi, and H. Toumi, "VM placement algorithm based on recruitment process within ant colonies," in 2016 International Conference on Digital Economy (ICDEc), pp. 1-7, 2016.
- [14] L. Zhang, Y. Wang, L. Zhu, and W. Ji, "Towards energy efficient cloud: an optimized ant colony model for virtual machine placement," Journal of Communications and Information Networks, Vol. 1, No. 4, pp. 116-132, 2016.
- [15] F. Alharbi, Y.-C. Tian, M. Tang, and M. H. Ferdous, "Profile-Based Ant Colony Optimization for Energy-Efficient Virtual Machine Placement," in International Conference on Neural Information Processing, pp. 863-871, 2017.
- [16] L. Zhu, R. Tang, Y. Tao, M. Ren, and L. Xue, "Multi-objective ant colony optimization algorithm based on load balance," in International Conference on Cloud Computing and Security, pp. 193-205, 2016.
- [17] X. Liu, H. Gu, H. Zhang, F. Liu, Y. Chen, and X. Yu, "Energy-aware on-chip virtual machine placement for cloud-supported cyber-physical systems," Microprocessors and Microsystems, Vol. 52, pp. 427-437, 2017.
- [18] S. M. Sait, A. Bala, and A. H. El-Maleh, "Cuckoo search based resource optimization of datacenters," Applied Intelligence, Vol. 44, No. 3, pp. 489-506, 2016.
- [19] C. Sonklin, M. Tang, and Y.-C. Tian, "A decrease-and-conquer genetic algorithm for energy efficient virtual machine placement in data centers," in 2017 IEEE 15th International Conference on Industrial Informatics (INDIN), pp. 135-140, 2017.

- [20] Y. Wu, M. Tang and W. Fraser, "A simulated annealing algorithm for energy efficient virtual machine placement," 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2012.
- [21] X. Li, Z. Qian, R. Chi, B. Zhang, and S. Lu, "Balancing resource utilization for continuous virtual machine requests in clouds," in 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, pp. 266-273, 2012.
- [22] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," Future generation computer systems, Vol. 28, No. 5, pp. 755-768, 2012.
- [23] A. Satpathy, S. K. Addya, A. K. Turuk, B. Majhi, and G. Sahoo, "A resource aware VM placement strategy in cloud data centers based on crow search algorithm," in 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1-6, 2017.
- [24] F. Ramezani, M. Naderpour, and J. Lu, "A multi-objective optimization model for virtual machine mapping in cloud data centres," in 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1259-1265, 2016.
- [25] A. Al-Moalimi, J. Luo, A. Salah, and K. Li, "Optimal Virtual Machine Placement Based on Grey Wolf Optimization", Electronics, Vol. 8, No. 3, p. 283, 297, 2019.
- [26] A. Fatima et al., "An Enhanced Multi-Objective Gray Wolf Optimization for Virtual Machine Placement in Cloud Data Centers," Electronics, Vol. 8, No. 2, p. 218, 2019.
- [27] W. Wei, H. Gu, W. Lu, T. Zhou, and X. Liu, "Energy Efficient Virtual Machine Placement With an Improved Ant Colony Optimization Over Data Center Networks," IEEE Access, Vol. 7, pp. 60617-60625, 2019.
- [28] Z. Li, Y. Li, T. Yuan, S. Chen, and S. Jiang, "Chemical reaction optimization for virtual machine placement in cloud computing," Applied Intelligence, Vol. 49, No. 1, pp. 220-232, 2019.
- [29] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," Advances in Engineering Software, Vol. 114, pp. 163-191, 2017.
- [30] R. Guddo, and K. Reddy, "Swarm-Inspired Task Scheduling Strategy in Cloud Computing," In Innovative Product Design and Intelligent Manufacturing Systems, pp. 743-751, 2020.
- [31] S. Gharepasha, M. Masdari, and A. Jafarian "Virtual machine placement in cloud data centers using a hybrid multi-verse optimization algorithm," Artificial Intelligence Review, pp. 1-37, 2020.
- [32] E. Parvizi, and M. Rezvani, "Utilization-aware energy-efficient virtual machine placement in cloud networks using NSGA-III meta-heuristic approach," Cluster Computing, pp. 1-23