

## A New Approach for Optimal Placement of Virtual Machines in Cloud Datacenters Using Discrete Gravitational Search Algorithm and Chaotic Functions

Sasan Gharehpasha<sup>1</sup>, Mohammad Masdari<sup>2\*</sup>, Ahmad Jafarian<sup>3</sup>

1- Department of Computer, Urmia Branch, Islamic Azad University, Urmia, Iran.

2\*- Department of Computer, Urmia Branch, Islamic Azad University, Urmia, Iran.

3- Department of Mathematics, Urmia Branch, Islamic Azad University, Urmia, Iran.

<sup>1</sup>s.gharehpasha@iaurmia.ac.ir, <sup>2\*</sup>m.masdari@iaurmia.ac.ir, and <sup>3</sup>a.jafarian@iaurmia.ac.ir

Corresponding author address: Mohammad Masdari, Faculty of Computer Engineering, Islamic Azad University, Urmia Branch, Urmia, Iran, Post Code : 63896 – 57169.

**Abstract-** Placement of virtual machines on physical machines in cloud computing infrastructure is an important issue. Our approach for placement of virtual machines includes a mapping process of this machines on physical machines in cloud datacenters. Optimal placement results in lower power consumption, optimal usage of resources, traffic reduce in datacenters, decrease in costs and also increase in functionality of datacenters in cloud datacenters. In this paper we propose a discrete gravitational search algorithm and chaotic function for placement of virtual machines on physical machines in cloud datacenters. Our primary goal for proposing the approach is minimizing resource wastage, power consumption and network links. At the end of this paper we also compare our results with some other metaheuristic algorithms. Our results show that this approach is more effective than previous algorithms.

**Keywords-** Gravitational Search Algorithm, Chaotic Functions, Cloud Computing, Virtualization.



## ارائه راهکاری جدید برای جای گذاری بهینه ماشین های مجازی در مراکز داده ابری با استفاده از الگوریتم جستجوی گرانشی گسسته و توابع آشوبناک

ساسان قره پاشا<sup>۱</sup>، محمد مصدری<sup>۲\*</sup>، احمد جعفریان<sup>۳</sup>

۱- گروه کامپیوتر، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران.

۲- گروه کامپیوتر، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران.

۳ گروه ریاضی، واحد ارومیه، دانشگاه آزاد اسلامی، ارومیه، ایران.

<sup>1</sup>s.gharehpasha@iaurmia.ac.ir, <sup>2\*</sup>m.masdari@iaurmia.ac.ir, and <sup>3</sup>a.jafarian@iaurmia.ac.ir

\* نشانی نویسنده مسئول: محمد مصدری، استان آذربایجان غربی، ارومیه، اول جاده سلماس، دانشگاه آزاد اسلامی واحد ارومیه، دانشکده فنی و مهندسی، کد پستی: ۵۷۱۶۹-۶۳۸۹۶.

چکیده- جای گذاری ماشین های مجازی بر روی ماشین های فیزیکی در ساختار ابری یکی از مهم ترین مسایل می باشد. مکان یابی ماشین های مجازی فرایندی را شامل می شود که در آن ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده ابری نگاشت پیدا می کنند. جای گذاری بهینه باعث کاهش مصرف انرژی، استفاده بهینه از منابع، کاهش ترافیک در مراکز داده ها، کاهش هزینه ها و همچنین افزایش کارایی مراکز داده در بستر ابری می شود. در این مقاله، ما الگوریتم فرا ابتکاری نیروی گرانشی را به صورت گسسته و با استفاده از توابع آشوبگر برای جای گذاری بهینه ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده ابری، پیشنهاد داده ایم. هدف از این کار به حداقل رساندن مصرف منابع، مصرف انرژی و تعداد لینک های ارتباطی می باشد. کارایی راهکار پیشنهادی با نتایج چندین الگوریتم فرا ابتکاری دیگر مقایسه می شود. نتایج نشان می دهد که الگوریتم پیشنهاد شده بهتر و تاثیرگذارتر از روش های مقایسه شده می باشد.

واژه های کلیدی: الگوریتم نیروی گرانشی، توابع آشوبگر، رایانش ابری، مجازی سازی.

### ۱- مقدمه

استفاده فراوان از وسایل هوشمند باعث افزایش استفاده از مدل ابری شده است. تعداد و اندازه مراکز داده ابری به سرعت در حال افزایش می باشد. با استفاده از محاسبات ابری هزینه ساختار شبکه و دخیله سازی به صورت پویا کاهش می یابد [۲ و ۳].

محاسبات ابری دارای یک گرایش عمومی نسبت به محاسبات دیگر می باشد و علت آن این می باشد که محاسبات ابری یک بستر ارزان و راحت برای دسترسی کاربران نسبت به منابع فراهم کرده است. محاسبات دارای بستری می باشد که منابع به صورت سرویس در اختیار کاربران قرار می گیرد. همچنین محاسبات ابری شرایطی

در سال های اخیر، محاسبات ابری به یک الگوی محاسباتی عمومی برای میزبانی و ارائه سرویس بر روی بستر اینترنت شده است [۱]. محاسبات ابری از مقیاس پذیری بالایی برخوردار بوده و سرویس های متنوعی را ارائه می دهد. امروزه، قابلیت پذیرش و استفاده از بستر ابری خیلی بالا بوده و هر روز میزان استفاده از این بستر افزایش می یابد. تاثیر استفاده از محاسبات ابری در زندگی روزمره بسیار بالا بوده و می توان از نمونه های استفاده از محاسبات ابری به شبکه های اجتماعی، شبکه های حسگر و غیره اشاره نمود.

*Archive of SID*

نهایت در بخش پنجم آزمایش‌ها و نتایج حاصل از آن و در بخش ششم نتیجه‌گیری حاصل از انجام این تحقیق مطرح می‌گردد.

## ۲- مروری بر کارهای پیشین

چاپ‌سیری و دوستان در [۷] از الگوریتم برنامه‌نویسی خطی برای جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری استفاده کرده‌اند. هدف در این مقاله، به حداقل رساندن تعداد ماشین‌های فیزیکی بوده است. برای رسیدن به این هدف از چندین فاکتور مهم استفاده شده است. در این مقاله سعی شده است که ماشین‌های مجازی را بر روی یک مجموعه از ماشین‌های فیزیکی که دارای ویژگی‌های یکسانی می‌باشند، تخصیص دهند. یکی دیگر از اهداف این مقاله، قرار گرفتن بیشترین ماشین مجازی بر روی یک ماشین فیزیکی می‌باشد که باعث کاهش مصرف انرژی، توان مصرفی و اتلاف انرژی می‌شود. با قرار گرفتن مناسب ماشین‌های مجازی بر روی ماشین‌های فیزیکی، تعداد انتقال و مهاجرت ماشین‌های مجازی نیز کاهش می‌یابد. با کاهش این اهداف، سرعت پاسخ‌دهی در مراکز داده افزایش می‌یابد.

مای و دوستان در [۸] از الگوریتم فرا ابتکاری ژنتیک بهبود یافته برای مکان‌یابی بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری استفاده کرده‌اند. در این مقاله، مهم‌ترین هدف به حداقل رساندن مصرف انرژی در مراکز داده ابری با مقیاس بزرگ بوده است. مصرف انرژی از لحاظ هزینه و افزایش آلاینده‌گی، یکی از مهم‌ترین نگرانی‌های امروزه می‌باشد. در این مقاله، سعی شده است با استفاده از الگوریتم ژنتیک تعداد ماشین‌های مجازی را بر روی ماشین‌های فیزیکی کمتری جای‌گذاری کرده و بدین ترتیب مصرف انرژی را در مراکز داده به حداقل برسانند. الگوریتم ژنتیک یکی از سریع‌ترین روش‌ها در بهینه‌سازی قرار دادن ماشین‌های مجازی بر روی ماشین‌های فیزیکی می‌باشد.

در مقاله‌ای دیگر گاو و همکاران نیز به بررسی جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده در بستر ابری پرداخته‌اند. در این تحقیق نیز برای به دست آوردن نتایج بهتر از الگوریتم‌های فرا ابتکاری به صورت چند هدفه استفاده شده است. در ابتدا به تعریفی از مجازی‌سازی در بستر ابری پرداخته، سپس اهداف را معلوم کرده و با فرمول‌سازی اهداف و استفاده از الگوریتم

را فراهم نموده‌اند تا کاربران بتوانند به راحتی از منابع به اشتراک‌گذاری شده در این بستر استفاده نمایند [۴].

مدل‌های استقرار رایانش ابری به صورت ابر خصوصی، عمومی، گروهی و ترکیبی طبقه‌بندی می‌شوند. اگر مدل رایانش ابری به صورت خصوصی باشد، آنگاه مالکیت ابر بر عهده یک سازمان و یا نهاد می‌باشد. نمونه‌ای از این مدل ابری، برای استفاده از نرم‌افزارهای کاربردی مالی و تجاری که حاوی اطلاعات حساس و حیاتی آن سازمان می‌باشد، کاربرد دارد. اگر رایانش ابری در حالت عمومی باشد، مالکیت ابر عمومی بر عهده‌ی ارائه دهنده‌ی آن خدمات است و منابع آن به عموم فروخته می‌شود. کاربران نهایی می‌توانند بخش‌هایی از منابع را اجاره کنند و معمولاً قادرند بر اساس نیاز خود منابع ارائه شده تحت ابر را مدیریت کنند. آمازون، گوگل و مایکروسافت نمونه‌هایی از ارائه دهندگان ابر عمومی می‌باشند. یکی دیگر از حالات رایانش ابری ابر گروهی می‌باشد. اگر محاسبات ابری به صورت رایانش ابری باشد آنگاه این امکان را فراهم می‌آورد که اعضای یک گروه از منابع آن استفاده کند و یا چند سازمان خصوصی که دارای اشتراکات داده‌ای می‌باشند، استفاده نمایند. آخرین حالت ابری در رایانش ابری، ابر ترکیبی می‌باشد. این ابر، ترکیبی از دو یا چند زیرساخت ابری می‌باشد که ممکن است به صورت خصوصی، عمومی یا گروهی باشند. هدف اصلی ابر ترکیبی معمولاً اختصاص منابع اضافی در مورد تقاضاهای بالا است [۵].

جای‌گذاری‌های ماشین‌های مجازی شامل عملیاتی برای نگاشت ماشین‌های مجازی بر روی ماشین‌های فیزیکی می‌باشد. قراردعی بهینه در رایانش ابری بسیار مهم بوده و باعث بهبود مصرف انرژی و مدیریت صحیح تخصیص منابع می‌شود. قراردعی ماشین‌های مجازی این امکان را در رایانش ابری فراهم می‌آورد که چندین ماشین مجازی بر روی یک میزبان فیزیکی قرار بگیرد. با توجه به افزایش روزافزون نیاز به محیط ابری و افزایش استفاده از سرویس‌های ابری، جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی تبدیل به یک مساله مهمی گردیده است [۶]. هدف این مقاله ارائه روشی برای بهینه‌سازی قرار دادن ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده ابری می‌باشد. در این راستا از ترکیب الگوریتم فرا ابتکاری جستجوی گرانشی گسسته با توابع آشوبناک استفاده شده است.

ادامه مقاله بدین صورت است که در بخش دوم برخی از پیشینه‌های تحقیق و در بخش سوم مفاهیم پایه مورد بحث قرار می‌گیرد. در بخش چهارم الگوریتم پیشنهادی ارائه می‌شود و در

## Archive of SID

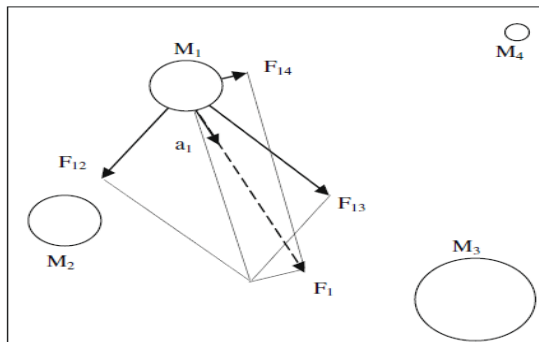
الگوریتم فرا ابتکاری جدید برای رفع این مشکل می‌باشد. یکی از این الگوریتم‌های فرا ابتکاری، الگوریتم نیروی جاذبه یا GSA می‌باشد. جاذبه تمایل نزدیک شدن اجسام به یکدیگر می‌باشد. جاذبه یکی از چهار تراکنش بنیادی در طبیعت می‌باشد. هر ذره در دنیا ذرات مجاور خود را تحت تاثیر می‌گذارد. جاذبه در همه جا وجود دارد. جاذبه بین دو ذره بدون هیچ واسطه و تاخیر انجام می‌شود. طبق قانون جاذبه نیوتن، هر ذره بر اساس قانون جاذبه ذره دیگر را تحت تاثیر قرار می‌دهد [۱۳]. قانون جاذبه مابین دو ذره رابطه مستقیم با نیروی گرانشی و حجم‌های ذرات و رابطه معکوس با مربع فاصله ذره‌ها دارد. رابطه ۱ قانون جاذبه را نشان می‌دهد.

$$F = G \frac{M_1 M_2}{R^2} \quad (1)$$

قانون دوم نیوتن می‌گوید که زمانی نیروی  $F$  به دره‌ای اعمال می‌شود، مقدار شتاب آن برابر با  $a$  می‌شود. شتاب حرکت یک ذره با نیروی  $F$  رابطه مستقیم و با حجم ذره رابطه غیر مستقیم دارد. رابطه ۲ میزان شتاب یک ذره را نشان می‌دهد.

$$a = \frac{F}{M} \quad (2)$$

شکل ۱، نیروی جاذبه و شتاب حرکت یک ذره را نسبت به ذرات دیگر نشان می‌دهد. همانطور که در شکل دیده می‌شود، ذرات بزرگتر نیروی بیشتری به ذرات کوچکتر وارد می‌کنند.



شکل ۱: نیروی جاذبه

تا به حال در مورد نحوه وارد آمدن جاذبه از ذرات مختلف به یکدیگر بر اساس قانون نیوتن اشاره شده بود. در این بخش در مورد نحوه اجرای الگوریتم جاذبه اشاره می‌شود. همانطور که اشاره شد این الگوریتم بر پایه و اساس جاذبه می‌باشد [۱۴]. در این الگوریتم ذرات به عنوان اشیا در نظر گرفته می‌شوند. کارایی آنها به وسیله حجم‌شان محاسبه می‌شود. همه اشیا به واسطه نیروی جاذبه روی یکدیگر تاثیر می‌گذارند. این تاثیر گذاشتن بر اساس نیروی جاذبه یک حرکت عمومی و جهانی را ما بین ذرات به وجود

فرا ابتکاری اقدام به حل مسئله جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی نموده‌اند [۹]. در این مقاله، از الگوریتم فرا ابتکاری ACO به صورت چند هدفه برای حل مساله جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی روی مراکز داده ابری استفاده شده است. با استفاده از این روش، سعی در به حداقل رساندن مصرف انرژی، اتلاف منابع و بالا رفتن سرعت پاسخ‌دهی می‌باشد.

یانگ و دوستان در [۱۰] الگوریتمی را برای جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مرکز داده ابری برای کاهش مصرف انرژی و کاهش ترافیک با استفاده از الگوریتم فرا ابتکاری ژنیتیک ارائه کرده‌اند. در این مقاله سعی شده است با جای‌گذاری بهینه ماشین‌های مجازی، تعداد مهاجرت را بر روی ماشین‌های فیزیکی کاهش داده که این کار باعث کاهش ترافیک در مرکز داده ابری می‌شد.

در [۱۱]، ویچائو و دوستان الگوریتم جدیدی را برای حل مسئله جای‌گذاری بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مرکز داده ابری ارائه داده‌اند. در این روش، از الگوریتم فرا ابتکاری کرم شب تاب چند هدفه استفاده شده است. مدیریت مصرف منابع، کاهش مصرف انرژی و افزایش کارایی مرکز داده ابری، معیارهایی بوده است که با استفاده از این روش مورد تحلیل قرار گرفته‌اند.

در [۱۲]، نویسندگان الگوریتمی برای به حداقل رساندن مصرف انرژی در مرکز داده ابری ارائه کرده بودند. آنها برای جای‌گذاری ماشین‌های مجازی در مراکز داده از الگوریتم PSO بهبود یافته استفاده کرده‌اند. در کل اهداف الگوریتم‌های ارائه شده می‌توان به صورت زیر بیان کرد.

- ۱- به حداقل رساندن مصرف انرژی و مدیریت منابع محاسباتی.
- ۲- به حداقل رساندن تعداد سرورهای فعال در مرکز داده ابری.
- ۳- به حداقل رساندن تعداد مهاجرت‌های ماشین‌های مجازی در مرکز داده ابری.

### ۳- مفاهیم پایه

#### ۳-۱- الگوریتم جستجوی گرانشی

در حل مسایل بهینه‌سازی با تعداد بعد زیاد در فضای جستجو، الگوریتم‌های بهینه‌سازی کلاسیک نتیجه قابل قبولی را ارائه نمی‌دهند. علت آن این است که فضای جستجو با بزرگ شدن اندازه مسئله، به شدت افزایش می‌یابد. بنابراین نیاز به یک سری

Archive of SID

طبق قانون دوم نیوتن، هر جسم در جهت بعد  $d$ ام شتابی می‌گیرد که متناسب است با نیروی وارد بر آن جسم در جهت  $d$ ام، بخش بر جرم اینرسی آن که در رابطه ۷ بیان شده است. شتاب جسم  $i$  در جهت بعد  $d$  در زمان  $t$  با  $a_i^d(t)$  و جرم اینرسی جسم  $i$  با  $M_{ii}$  نشان داده شده است.

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (7)$$

سرعت بعدی هر جسم برابر مجموع ضربی از سرعت فعلی جسم و شتاب جسم تعریف می‌شود که در رابطه ۸ نمایش داده شده است و همچنین موقعیت جدید بعد  $d$  از جسم  $i$  طی رابطه ۹ محاسبه می‌شود.  $v_i^d(t)$  سرعت بعد  $d$  جسم  $i$  در زمان  $t$  است.

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (8)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (9)$$

برای تنظیم ثابت گرانش، از یک مقدار اولیه شروع کرده، با گذشت زمان مقدار آن کاهش داده می‌شود. ثابت گرانش طبق رابطه ۱۰، تابعی از ثابت گرانش اولیه و زمان است. این موضوع در دنیای واقعی نیز صدق می‌کند و ثابت گرانش با آهنگ بسیار کمندی در طول زمان کوچک می‌شود. در نسخه پیوسته این الگوریتم، یک پیشنهاد برای این تابع، استفاده از رابطه نمایی جهت کاهش ثابت گرانش می‌باشد. رابطه فوق به صورت ۱۱ قابل بیان است. لازم به ذکر است در کاربردهای مختلف، می‌توان از روابط مناسب دیگری استفاده کرد. در کاربردهای خاص، می‌توان با تعریف روابط مناسب جهت ثابت گرانش، کارایی الگوریتم را بالا برد.

$$G(t) = G(G_0, t) \quad (10)$$

$$G(t) = G_0 e^{-\alpha \frac{t}{T}} \quad (11)$$

در رابطه ۱۱،  $G_0$  ثابت گرانش اولیه،  $\alpha$  یک ثابت مثبت و  $T$  کل تکرارهای الگوریتم و به عبارتی طول عمر سیستم است. در این الگوریتم، اجرام گرانشی و اینرسی مطابق رابطه ۱۲، برابر در نظر گرفته شده، برای تنظیم آنها، از مقدار تابع هدف اجسام با استفاده از رابطه ۱۳ استفاده می‌شود. مقدار اجرام در رابطه ۱۴، نرمالیزه می‌شود. در این روابط، به اجسام با شایستگی بهتر، جرم بیشتری نسبت داده شود.

$$M_{ai} = M_{pi} = M_{ij} = M_i, \quad i = 1, 2, \dots, N \quad (12)$$

$$q_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (13)$$

می‌آورد و ذرات کوچکتر به طرف ذرات بزرگتر حرکت می‌کنند. ذرات با حجم سنگین‌تر که نتیجه مناسبی می‌باشند، با سرعت کمتری نسبت به ذرات سبک‌تر حرکت می‌کنند. این معنای استخراج را در الگوریتم تضمین می‌کند.

مکان ذره به راهکار مسئله بستگی دارد و ما بقیه موارد در تابع برازندگی تعیین می‌گردند. به عبارتی دیگر، هر ذره یک راهکار را نمایش می‌دهد و الگوریتم با تنظیم کردن مناسب و درست نیروی جاذبه و اینرسی ذرات هدایت می‌شود. در این الگوریتم انتظار می‌رود که ذرات تاثیر زیادی از ذرات سنگین‌تر داشته باشند. این ذرات به عنوان جواب‌های بهینه در فضای جستجو در نظر گرفته می‌شوند. الگوریتم فرا ابتکاری GSA به صورت یک سیستم ایزوله از ذرات در نظر گرفته می‌شود. مانند یک دنیای مصنوعی بوده که در این دنیا فقط ذرات وجود دارند و دو قانون جاذبه و حرکت در آن وجود دارند.

فضای جستجو را با  $N$  ذره در نظر بگیرید. مکان هر ذره به صورت رابطه ۳ می‌باشد.

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^m) \quad \text{for } i=1, 2, \dots, N \quad (3)$$

در این سیستم، در زمان  $t$  به ذره  $i$  از سوی ذره  $j$  در جهت بعد  $d$  نیرویی به اندازه  $F_{ij}^d(t)$  وارد می‌شود. مقدار این نیرو در رابطه ۴ محاسبه می‌شود.  $M_{pi}$  و  $M_{aj}$  به ترتیب جرم گرانشی فعال جسم  $j$  و جرم گرانشی غیر فعال جسم  $i$  می‌باشند،  $G(t)$  ثابت گرانش در زمان  $t$  و  $R_{ij}$  فاصله بین دو جسم  $i$  و  $j$  می‌باشند. برای تعیین فاصله بین اجسام مطابق رابطه ۵ از فاصله اقلیدسی استفاده شده است.  $\epsilon$  یک عدد بسیار کوچک است.  $p$  توان فاصله است که یک عدد حقیقی بزرگتر از یک می‌باشد. این مقدار غالباً برابر یک در نظر گرفته شده است.

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t)^p + \epsilon} (x_j^d(t) - x_i^d(t)) \quad (4)$$

$$R_{ij}(t) = \|X_i(t), X_j(t)\|_2 \quad (5)$$

نیروی وارد بر جسم  $i$  در جهت بعد  $d$  در زمان  $t$  ( $F_i^d(t)$ )، مطابق رابطه ۶ برابر مجموع ضرب‌های تصادفی نیروهایی است که  $K$  جسم برتر بر جسم وارد می‌کنند. در این رابطه یک عدد تصادفی با توزیع یکنواخت در بازه  $[0-1]$  است.

$$F_i^d(t) = \sum_{j \in Kbest, j \neq i} rand_j^d F_{ij}^d(t) \quad (6)$$

## Archive of SID

ماشین های فیزیکی می باشند، قرار می گیرند. ماشین فیزیکی دارای یکسری منابع ای می باشد که اگر جای گذاری ماشین های 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 منبع ای هدر نمی رود. در این حالت یک ماشین فیزیکی دیگر وظیفه میزبانی را برای ماشین های مجازی بر عهده می گیرد. اگر جای گذاری ماشین های مجازی به صورت بهینه انجام نشود، منابع سخت افزاری ماشین های فیزیکی در مرکز داده ابری هدر می رود. با جای گذاری بهینه ماشین های مجازی می توان از توان پردازشی ماشین های فیزیکی به صورت کامل استفاده نمود که باعث جلوگیری از هدر رفتن منابع سخت افزاری در مرکز داده ابری می شود. برای داشتن یک مرکز داده ابری هوشمند، باید مصرف منابع در ماشین های فیزیکی به حداقل برسد.

در این مقاله از ترکیب الگوریتم فرا ابتکاری جستجوی گرانشی به صورت گسسته با توابع آشوبناک برای حل مسئله جای گذاری ماشین های مجازی بر روی ماشین های فیزیکی در مراکز داده ابری استفاده شده است.

جای گذاری ماشین های مجازی بر روی ماشین های فیزیکی یک مسئله گسسته می باشد. نحوه عملکرد الگوریتم های فرا ابتکاری به صورت پیوسته انجام می شود. بنابراین برای انجام جای گذاری ماشین های مجازی بر روی ماشین های فیزیکی در مرکز داده ابری از عملگرهای جدیدی استفاده می کنیم که مسئله جای گذاری را به صورت گسسته با الگوریتم پیشنهادی حل نماید.

در روش پیشنهادی، عملگرهای جدید شامل عملگرهای ضرب، منها و جمع می باشد. در الگوریتم پیشنهادی از یک عملگر جدید منها، به جای جمع از عملگر برش و به جای عملگر ضرب از عملگر جهش استفاده خواهیم کرد. جای گذاری شامل عملیاتی برای نگاشت ماشین های مجازی بر روی ماشین های فیزیکی می باشد. ماشین های مجازی به این صورت روی ماشین های فیزیکی قرار می گیرند.

ماشین مجازی	VM1	VM2	VM3	VM4	VM5
ماشین فیزیکی	PM1	PM2	PM1	PM1	PM2

شکل ۲: نگاشت ماشین های مجازی

$$M_i(t) = \frac{q_i(t)}{\sum_{j=1}^N q_j(t)} \quad (14)$$

در این روابط  $fit_i(t)$  بیانگر میزان برازندگی جسم  $i$  در زمان  $t$  است. در مسایل کمینه یابی می توان از روابط ۱۵ و ۱۶ برای محاسبه بهترین و بدترین مقدار شایستگی استفاده کرد. در مسایل بیشینه یابی بهترین و بدترین طبق روابط ۱۷ و ۱۸ تعریف می شود.

$$best(t) = \min_{j \in \{1, \dots, N\}} fit_j(t) \quad (15)$$

$$worst(t) = \max_{j \in \{1, \dots, N\}} fit_j(t) \quad (16)$$

$$best(t) = \max_{j \in \{1, \dots, N\}} fit_j(t) \quad (17)$$

$$worst(t) = \min_{j \in \{1, \dots, N\}} fit_j(t) \quad (18)$$

در این مقاله از الگوریتم جستجوی گرانشی به صورت گسسته استفاده خواهیم کرد.

### ۲-۳- توابع آشوبگر

الگوریتم های بهینه سازی که از توابع آشوبگر استفاده می کنند، نوعی خاصی از الگوریتم های بهینه سازی بر پایه متغیرهای تصادفی می باشد. این الگوریتم ها از متغیرهای آشوبناک به جای متغیرهای تصادفی استفاده می کنند. این متغیرها دارای بی نظمی خاصی می باشند. این متغیرها دارای یک ویژگی خاصی بوده و آن این است که هنگام توزیع، تکرار ناپذیر می باشند. بنابراین با توجه به این ویژگی، الگوریتم هایی که از متغیرهای آشوبناک استفاده می کنند یک جستجوی کامل را در فضای مساله نسبت به جستجوهای دیگر انجام می دهند. توابع آشوبگر متعددی وجود دارد، دو تابع آشوبگر Logistic و Chebyshev در این مقاله استفاده شده است. روابط ۱۹ و ۲۰ بیانگر این توابع می باشند.

$$X_{n+1} = r X_n (1 - X_n) \quad (19)$$

$$X_{k+1} = \cos(k \cos^{-1}(X_k)) \quad (20)$$

### ۴- الگوریتم پیشنهادی

یک مرکز داده ابری با چندین نوع ماشین فیزیکی با سخت افزارهای متفاوت و چندین نوع ماشین مجازی مختلف با نیازمندی های متفاوت وجود دارد. ماشین های مجازی موجود در مرکز داده ابری برای شروع کار خود بر روی میزبان هایی که همان

*Archive of SID*

این خانه‌ها از ذره تغییر می‌یابد. بدین ترتیب در الگوریتم پیشنهادی از سه عملگر جدید استفاده می‌شود.

در الگوریتم پیشنهادی، از توابع و متغیرهای آشوبناک، برای نتایج بهتر استفاده شده است. بنابراین در الگوریتم پیشنهادی، الگوریتم جستجوی گرانشی به واسطه متغیرهای آشوبناک و با استفاده از عملگرهای جدید بهبود داده شده است. شبه کد الگوریتم پیشنهادی در شکل شماره ۵ و همچنین دیاگرام بلوکی الگوریتم پیشنهادی در شکل شماره ۶ نشان داده شده است.

**Generate initial population**

**Evaluate the fitness for each agent**

**While** (the end criterion is not satisfied)

**Update**  $G$  by chaotic maps

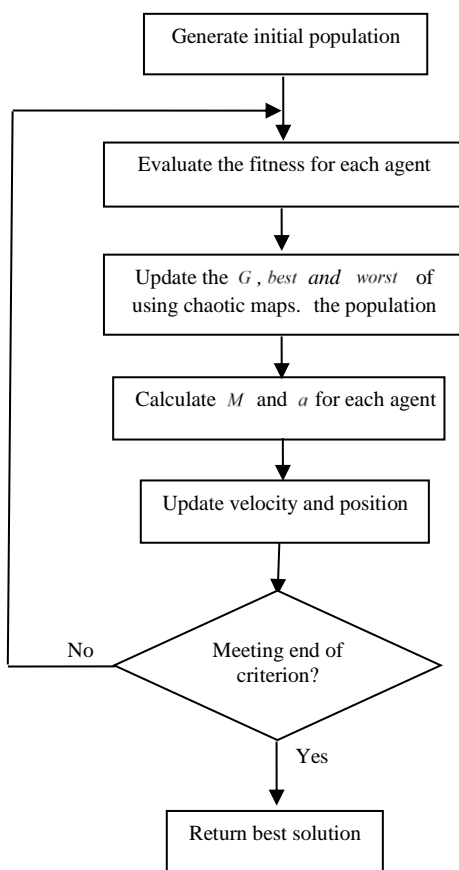
**Update**  $M$ , forces, and accelerations

**Update** velocities and positions

**End while**

**Return** the best search agent

شکل ۵: شبه کد الگوریتم پیشنهادی



در الگوریتم پیشنهادی از یک عملگر منهای جدید استفاده شده است. ساختار دو ذره A و ذره B در شکل شماره ۳ نشان داده شده است.

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

شکل ۳: 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. 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 شماره ۴ نحوه عملکرد این عملگر را نمایش می‌دهد.

A	1	4	3	1	2
B	2	3	1	2	3

نقطه برش اول      نقطه برش دوم

A	1	3	1	2	2
B	2	4	3	1	3

شکل ۴: عملگر جمع

در شکل شماره ۴، یکی از ذرات A و B به صورت تصادفی به عنوان جواب انتخاب می‌شود.

در الگوریتم پیشنهادی، به جای عملگر ضرب از عملگر جهش استفاده شده است. در این عملگر یک عدد به طور تصادفی انتخاب شده و به اندازه عدد تصادفی انتخاب شده، تعداد خانه به صورت تصادفی از ذره انتخاب شده و محتوای

مراکز داده دارای دو نوع ترافیک می باشد، ترافیک ما بین ماشین های مجازی و ترافیک ما بین ماشین های مجازی با ماشین های فیزیکی. میزان ترافیک از طریق رابطه ۲۳ نمایش داده می شود [۱۵].

$$T_{cost} = \sum_{i,j=1}^{n_i} a_{ij} b_{\pi(i)*\pi(j)} + \sum_{i=1}^{n_i} e_i g_{\pi(i)} \quad (23)$$

در رابطه ۲۳، قسمت اول نشان دهنده ترافیک ما بین ماشین های مجازی و قسمت دوم نشان دهنده ترافیک ماشین های مجازی با ماشین های فیزیکی می باشد. تاخیر ارسال نشان دهنده مدت زمانی می باشد که بسته از مبدا تا مقصد طول می کشد، که برسد. میزان تاخیر انتها به انتها با استفاده از رابطه ۲۴ محاسبه می شود [۱۰].

$$d_{end-end} = N[ d_{trans} + d_{prop} + d_{proc} + d_{queue}] \quad (24)$$

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 از ۲۴ سرور ناهمگن در مرکز داده رایانش ابری استفاده شده است. هر ماشین فیزیکی دارای یک منبع سخت افزاری با مشخصات، CPU با کارایی MIPS ۱۰۰۰، MIPS ۲۰۰۰ و MIPS ۳۰۰۰، دارای ۱۰ گیگابایت حافظه، ۱ گیگابایت پهنای باند و ۱ ترا بایت محل ذخیره سازی می باشد.

در این مقاله سه سری شبیه سازی با دو معماری VL2 و Three Tier و با تعداد ۳۲، ۴۸، ۶۴ ماشین های مجازی انجام شده است. هر ماشین مجازی دارای یک CPU با کارایی MIPS ۲۵۰، MIPS ۵۰۰، MIPS ۷۵۰ و MIPS ۱۰۰۰، دارای ۱۲۸ مگابایت حافظه، ۲۵۰ مگابایت پهنای باند و ۲۵۰۰ مگابایت محل ذخیره سازی می باشد. نتایج شبیه سازی با الگوریتم پیشنهادی با الگوریتم های فرا ابتکاری GSA، SOS، SFLA، FCFS، BBO مقایسه خواهد شد. نتایج شبیه سازی به صورت زیر می باشد. جداول ۱ و ۲ نتایج جای گذاری ۳۲ ماشین مجازی بر روی ماشین های فیزیکی را نشان می دهد.

جدول ۱: نتایج جای گذاری ۳۲ ماشین مجازی در ماشین فیزیکی در

معماری VL2

الگوریتم	مدیریت		تاخیر	
	مصرف انرژی	منابع	پهنای باند	ارسال
Logistic GS	۲۲۳۲,۸۷۴	۲,۰۶۴۹۱۲	۱۳۰۲,۶	۴,۳۲
Chebyshev GSA	۲۲۳۰,۷۵۲	۱,۹۸۶۸۶۳	۱۳۰,۲	۴,۳

شکل ۶: دیاگرام بلوکی الگوریتم پیشنهادی

## ۵- 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 تحقیقات اخیر نشان داده شده است که مصرف انرژی توسط سرورهای مراکز داده می تواند به صورت دقیق به وسیله یک رابطه خطی مابین مصرف انرژی و مصرف CPU اندازه گیری شود. رابطه ۲۱ مصرف انرژی را در مراکز داده رایانش ابری نشان می دهد.

$$P_j = P_{idle} + (P_{busy} - P_{idle} * U_j) \quad (21)$$

در رابطه ۲۱،  $P_j$  نشان دهنده مصرف انرژی زام ماشین فیزیکی در مرکز داده می باشد.  $U_j$  میزان قدرت CPU ماشین فیزیکی بر حسب MIPS می باشد.

یکی دیگر از فاکتورهای مهم در مساله جای گذاری ماشین های مجازی، جلوگیری از اتلاف انرژی در مراکز داده رایانش ابری می باشد. هر سرور موجود در مرکز داده دارای یک منبع سخت افزاری بوده و ماشین های مجازی مختلفی را می تواند میزبانی کند. منابع استفاده نشده بر روی سرورها باید به صورت مناسب مدیریت شود. رابطه ۲۲ مقدار اتلاف انرژی را نشان می دهد.

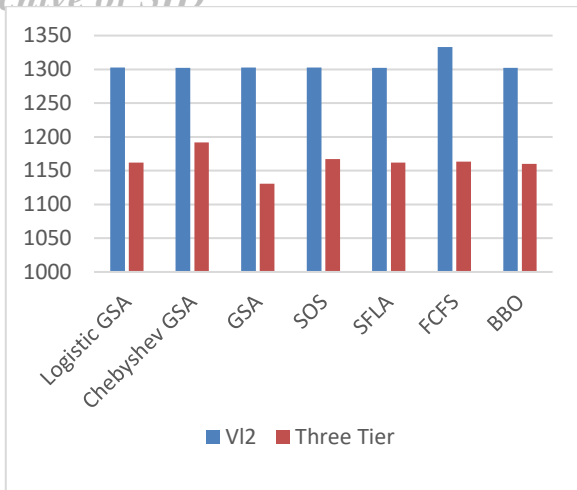
$$W_j = \frac{|L_j^p - L_j^m| + \epsilon}{U_j^p + U_j^m} \quad (22)$$

در رابطه ۲۲،  $W_j$  میزان اتلاف انرژی را نشان می دهد.  $U_j^m$  و  $U_j^p$  میزان مصرف CPU و حافظه سرور را در بعد ژام نمایش می دهند و همچنین  $L_j^m$  و  $L_j^p$  میزان بیکاری CPU و حافظه را بر روی سرور در بعد ژام نشان می دهند.

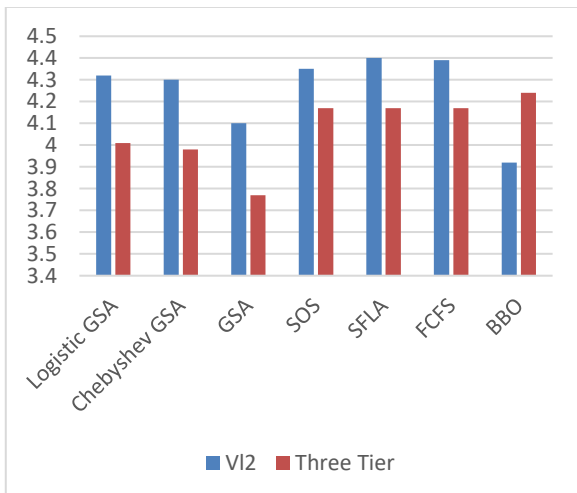
میزان ترافیک انتقالی نیز در رایانش ابری نیز بسیار مهم بوده است. هدف به حداقل رساندن ترافیک در رایانش ابری می باشد. در



Archive of SID



شکل ۹: نمودار پهنای باند برای ۳۲ ماشین مجازی با معماری VL2 و Three Tier



شکل ۱۰: نمودار تاخیر ارسال برای ۳۲ ماشین مجازی با معماری VL2 و Three Tier

جداول ۳ و ۴ نتایج جای‌گذاری ۴۸ ماشین مجازی بر روی ماشین‌های فیزیکی را در محیط ابری نشان می‌دهند.

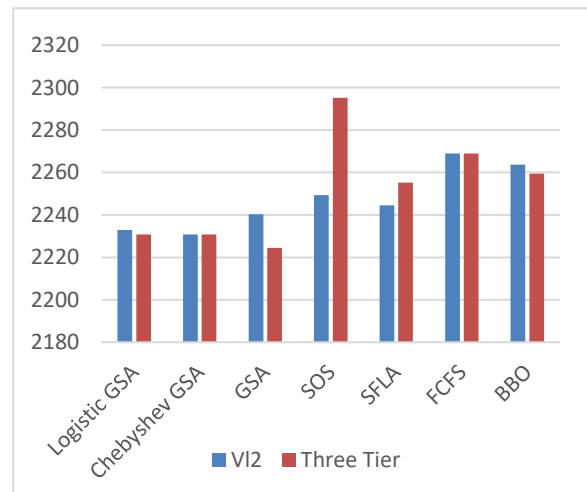
جدول ۳: نتایج جای‌گذاری ۴۸ ماشین مجازی در ماشین فیزیکی در معماری VL2

الگوریتم	مصرف انرژی	مدیریت منابع	پهنای باند	تأخیر ارسال
Logistic GSA	۲۳۷۳.۶۹۵	۰.۹۹۷۲۷۶	۱۳۰۳.۲	۶.۱۲
Chebyshev GSA	۲۳۱۷.۵۷۲	۰.۹۴۷۸۳۷	۱۳۳۳.۲	۶.۳۲
GSA	۲۳۷۳.۲۶۷	۱.۰۶۲۶۴۶	۱۳۰۳.۲	۵.۹۲
SOS	۲۳۹۱.۷۷۲	۱.۱۷۵۱۸۱	۱۳۰۳.۸	۶.۳۷
SFLA	۲۳۹۰.۷۲۰	۰.۹۷۵۱۸۱	۱۳۰۳.۲	۶.۳۵
FCFS	۲۳۹۰.۶۷۲	۰.۹۵۱۸۱۶	۱۳۰۳.۲	۶.۳۵
BBO	۲۳۷۳.۲۶۷	۱.۱۸۸۲۴	۱۳۳۳.۸	۶.۴۱

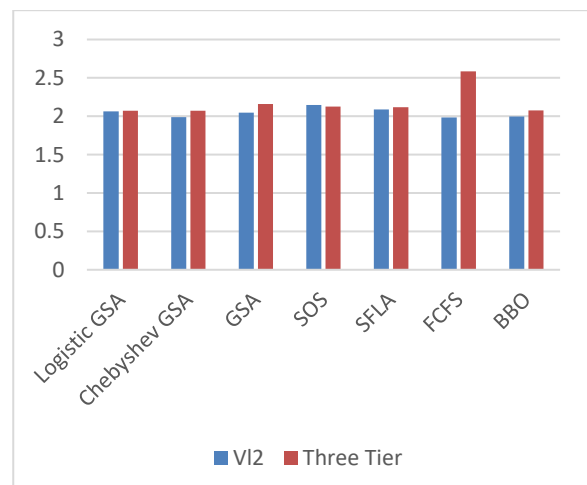
GSA [14]	۲۲۴۰.۳۰۲	۲.۰۴۸۸۵۲	۱۳۰۳.۶	۴.۱
SOS [16]	۲۲۴۹.۳۰۲	۲.۱۴۸۸۵۲	۱۳۰۳.۸	۴.۳۵
SFLA [17]	۲۲۴۴.۵۴۶	۲.۰۸۷۸۲۶	۱۳۰۳	۴.۴
FCFS [18]	۲۲۶۸.۹۵۲	۱.۹۸۵۸۱۸	۱۳۳۳.۲	۴.۳۹
BBO [19]	۲۲۶۳.۶۴۶	۱.۹۹۷۵۶۱	۱۳۳۲	۳.۹۲

جدول ۲: نتایج جای‌گذاری ۳۲ ماشین مجازی در ماشین فیزیکی در معماری Three Tier

الگوریتم	مصرف انرژی	مدیریت منابع	پهنای باند	تأخیر ارسال
Logistic GSA	۲۲۳۰.۷۵۲	۲.۰۶۹۸۶۳	۱۱۶۲	۴.۰۱
Chebyshev GSA	۲۲۳۰.۷۵۲	۲.۰۶۹۸۶۳	۱۱۹۲	۳.۹۸
GSA	۲۲۲۴.۳۸۶	۲.۱۶۱۳۸۵	۱۱۳۰.۸	۳.۷۷
SOS	۲۲۹۵/۱۵۸	۲.۱۳۶۰۹۷	۱۱۶۷	۴.۱۷
SFLA	۲۲۵۵.۱۵۸	۲.۱۱۶۰۹۷	۱۱۶۲	۴.۱۷
FCFS	۲۲۶۸.۹۵۲	۲.۵۵۵۸۱۸	۱۱۶۳.۲	۴.۱۷
BBO	۲۲۵۹.۴۰۲	۲.۰۷۶۸۲۹	۱۱۶۰.۲	۴.۲۴

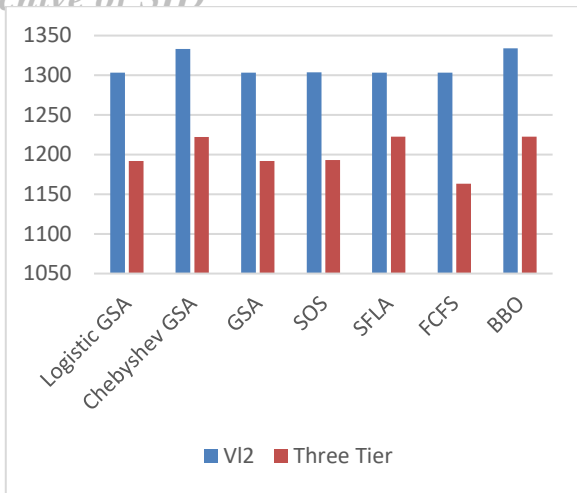


شکل ۷: نمودار مصرف انرژی برای ۳۲ ماشین مجازی با معماری VL2 و Three Tier

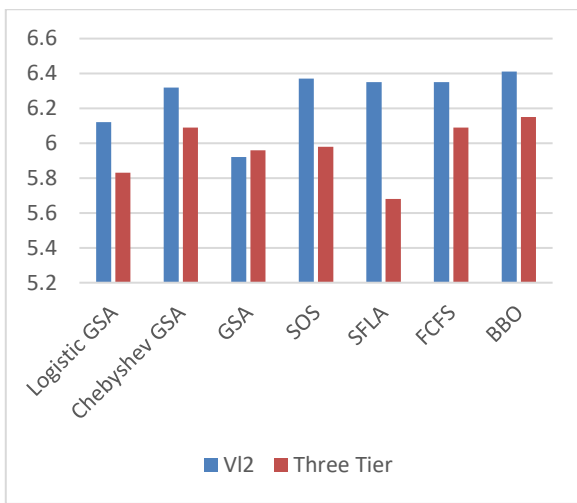


شکل ۸: نمودار مدیریت منابع برای ۳۲ ماشین مجازی با معماری VL2 و Three Tier

Archive of SID



شکل ۱۳: نمودار پهنای باند برای ۴۸ ماشین مجازی با معماری VL2 و Three Tier



شکل ۱۴: نمودار تاخیر ارسال برای ۴۸ ماشین مجازی با معماری VL2 و Three Tier

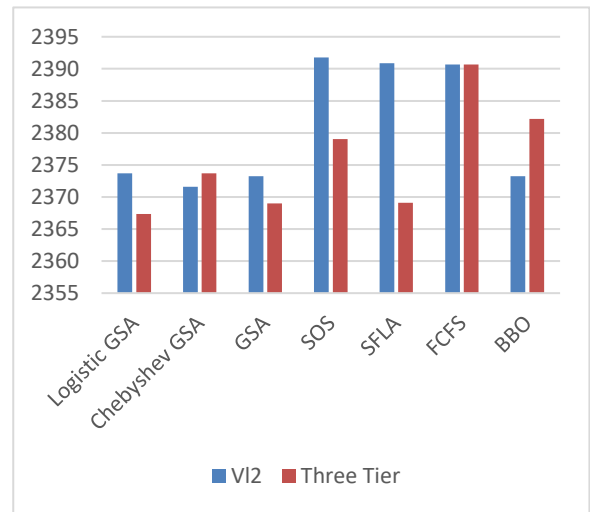
جداول ۵ و ۶ نتایج جای گذاری ۶۴ ماشین مجازی بر روی ماشین های فیزیکی را در محیط ابری نشان می دهند.

جدول ۵: نتایج جای گذاری ۶۴ ماشین مجازی در ماشین فیزیکی در معماری VL2

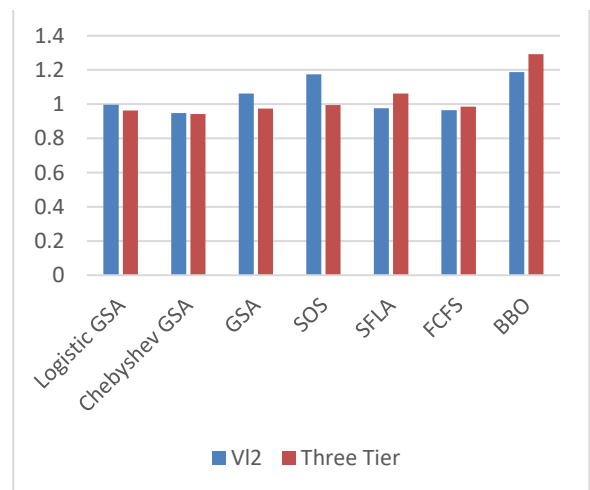
الگوریتم	مصرف انرژی	مدیریت منابع	پهنای باند	تأخیر ارسال
Logistic GSA	۲۴۷۳,۱۳۲	۰,۷۸۱۴۶۲	۱۳۳۳,۸	۱۰,۶۸
Chebyshev GSA	۲۴۸۰,۵۵۹	۰,۸۲۲۸۱۱	۱۳۳۳,۸	۹,۷۴
GSA	۲۴۹۰,۱۰۹	۰,۷۹۸۷۸۳	۱۳۳۲,۶	۱۰,۰۶
SOS	۲۴۸۱,۳۳۷	۰,۷۹۵۳۶	۱۳۳۳,۸	۱۰,۱۲
SFLA	۲۴۷۹,۳۳۷	۰,۸۸۰۳۶	۱۳۳۳,۸	۱۰,۰۲
FCFS	۲۵۰۳,۹۰۴	۰,۸۴۰۱۶۳	۱۳۳۳,۸	۱۰,۰۹
BBO	۲۴۹۲,۳۳۲	۰,۹۷۸۵۱۹	۱۳۳۳,۸	۱۰,۰۹

جدول ۴: نتایج جای گذاری ۴۸ ماشین مجازی در ماشین فیزیکی در معماری Three Tier

الگوریتم	مصرف انرژی	مدیریت منابع	پهنای باند	تأخیر ارسال
Logistic GSA	۲۳۶۷,۳۲	۰,۹۶۳۲۵۳	۱۱۹۲	۵,۸۳
Chebyshev GSA	۲۳۷۳,۶۹۵	۰,۹۴۲۸۸۶	۱۲۲۲	۶,۰۹
GSA	۲۳۶۹,۰۲۲	۰,۹۷۴۹۹۶	۱۱۹۲	۵,۹۶
SOS	۲۳۷۹,۰۲۲	۰,۹۹۴۹۹۶	۱۱۹۳	۵,۹۸
SFLA	۲۳۶۹,۰۸۴	۱,۰۶۲۸۵۷	۱۲۲۲,۶	۵,۶۸
FCFS	۲۳۹۰,۶۷۲	۰,۹۸۵۱۸۱	۱۱۶۳,۲	۶,۰۹
BBO	۲۳۸۲,۱۸۳	۱,۲۹۱۵۴۵	۱۲۲۲,۶	۶,۱۵

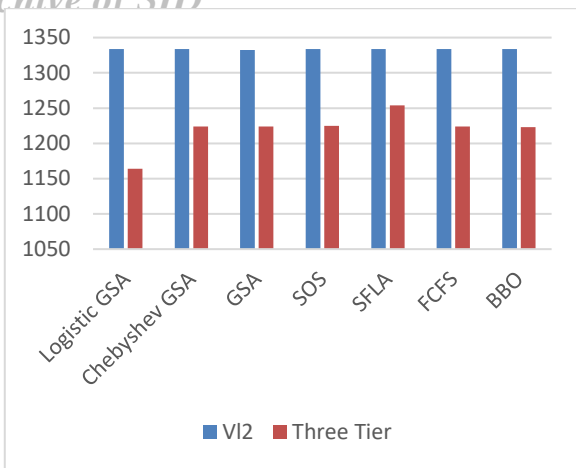


شکل ۱۱: نمودار مصرف انرژی برای ۴۸ ماشین مجازی با معماری VL2 و Three Tier

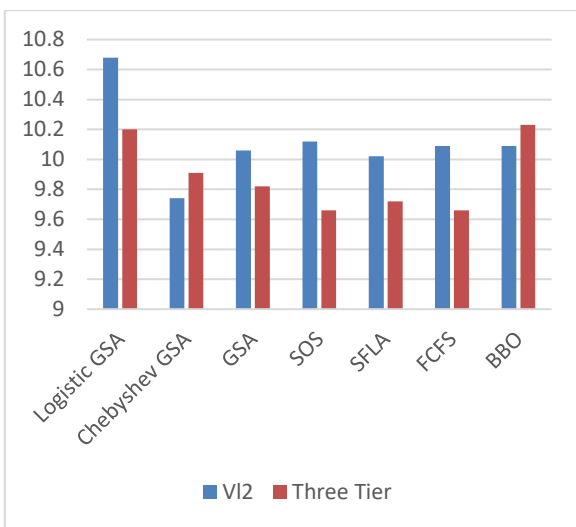


شکل ۱۲: نمودار مدیریت منابع برای ۴۸ ماشین مجازی با معماری VL2 و Three Tier

Archive of SID



شکل ۱۷: نمودار پهنای باند برای ۶۴ ماشین مجازی با معماری VL2 و Three Tier



شکل ۱۸: نمودار تاخیر ارسال برای ۶۴ ماشین مجازی با معماری VL2 و Three Tier

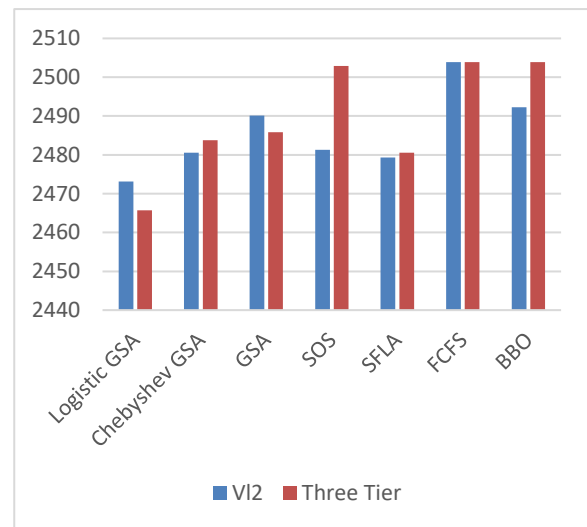
۶- نتیجه گیری و کارهای آتی

ما در این مقاله، به بررسی چگونگی قرار گرفتن بهینه ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده رایانش ابری پرداخته و مساله جای‌گذاری ماشین‌های مجازی را بر روی ماشین‌های فیزیکی با استفاده از الگوریتم پیشنهادی حل کرده و نتایج آن را با سایر الگوریتم‌های فرا ابتکاری مقایسه و بررسی کرده‌ایم. نتایج به دست آمده نشان دهنده این است که الگوریتم پیشنهادی در بسیاری از موارد بهتر از نتایج سایر الگوریتم‌ها عمل کرده است. در تحقیقات بعدی سعی خواهد شد که با روش‌های جدید نتایج بهتری در جای‌گذاری ماشین‌های مجازی بر روی ماشین‌های فیزیکی در مراکز داده رایانش ابری، حاصل شود.

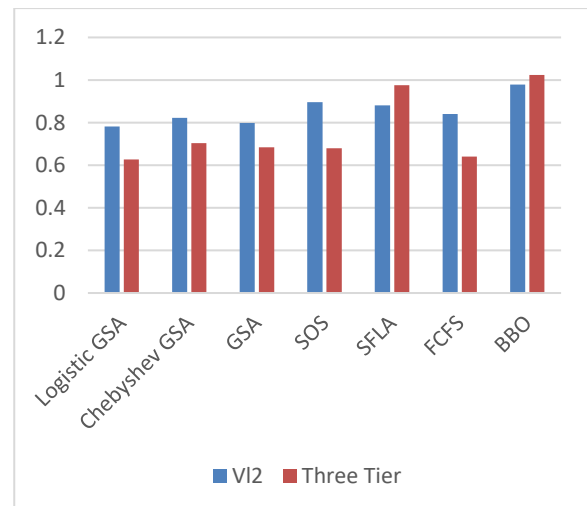
جدول ۶: نتایج جای‌گذاری ۶۴ ماشین مجازی در ماشین فیزیکی در

معماری Three Tier

الگوریتم	مصرف انرژی	مدیریت منابع	پهنای باند	تأخیر ارسال
Logistic GSA	۲۴۶۵,۷۰۴	۰,۶۲۷۰۶۶	۱۱۶۳,۸	۱۰,۰۲
Chebyshev GSA	۲۴۸۳,۷۴۳	۰,۷۰۳۴۹	۱۲۲۳,۸	۹,۹۱
GSA	۲۴۸۵,۸۶۵	۰,۶۸۳۶۹۷	۱۲۲۳,۸	۹,۹۸
SOS	۲۵۰۲,۹۰۴	۰,۶۸۰۱۶۳	۱۲۲۴,۸	۹,۶۶
SFLA	۲۴۸۰,۵۵۹	۰,۹۷۵۴۴۶	۱۲۵۳,۸	۹,۷۲
FCFS	۲۵۰۲,۹۰۴	۰,۶۴۰۱۶۳	۱۲۲۳,۸	۹,۶۶
BBO	۲۵۰۲,۹۰۴	۱,۰۲۳۴۹	۱۲۲۳,۲	۱۰,۲۳



شکل ۱۵: نمودار مصرف انرژی برای ۶۴ ماشین مجازی با معماری VL2 و Three Tier



شکل ۱۶: نمودار مدیریت منابع برای ۶۴ ماشین مجازی با معماری VL2 و Three Tier

- [1] A. Beloglazov, R. Buyya, "Energy efficient resource management in virtualized cloud data centers," Proceedings of the 2010 IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010, pp. 826-831.
- [2] G. Lee, "Cloud Networking: Understanding Cloud Based Data Center Networks," Morgan Kaufmann, 2014.
- [3] Q. Zhang, L. Cheng, R. Boutaba, "Cloud computing: state-of-the-art and research challenges," Journal of Internet Services and Applications, Vol. 1, No. 1, pp.7-18, 2010.
- [4] Y. Fang, D. Tang, J. Ge, "Energy-aware schedule strategy based on dynamic migration of virtual machines in cloud computing," Journal of Computational Information Systems, Vol. 10, No. 8, pp. 201-208, 2012.
- [5] M. Alicherry, T. Lakshman, "Optimizing data access latencies in cloud systems by intelligent virtual machine placement," 2013 Proceedings IEEE INFOCOM, 2013, pp. 647-655.
- [6] D. Kusic, J. Kephart, J. Hanson, N. Kandasamy, G. Jiang, "Power and performance management of virtualized computing environments via lookahead control," Cluster Computing, Vol. 12, No. 1, pp. 1-15, 2009.
- [7] S. Chaisiri, B. Lee, D. Niyato, "Optimal virtual machine placement across multiple cloud providers," Proceedings of the IEEE Asia-Pacific Services Computing Conference, 2009, pp. 103-110.
- [8] H. Mi, H. Wang, G. Yin, Y. Zhou, D. Shi, L. Yuan, "Online self-reconfiguration with performance guarantee for energy-efficient large-scale cloud computing data centers," Proceedings of the IEEE International Conference on Services Computing, 2010, pp. 514-521.
- [9] Y. Gao, H. Guan, Z. Qi, Y. Houb, L. Liu, "A multi objective ant colony system algorithm for virtual machine placement in cloud computing," Journal of Computer and System Sciences, Vol. 7, No. 9, pp. 1230-1242, 2013.
- [10] T. Yang, Y. Choon, A. Zomaya, "Energy-Efficient Data Center Networks Planning with Virtual Machine Placement and Traffic Configuration," 2014 IEEE 6th International Conference on Cloud Computing Technology and Science, 2014, pp. 284-291.
- [11] W. Ding, C. Gu, F. Luo, Y. Chang, U. Rugwiro, X. Li, G. Wen, "DFA-VMP: An efficient and secure virtual machine placement strategy under cloud environment," The Institute of Electronics, Information and Communication Engineers Vol. 101, No. 7, pp. 203-206, 2018.
- [12] E. Dashti, A. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing," Journal of Experimental & Theoretical Artificial Intelligence, Vol. 28, No. 1, pp. 97-112, 2016.
- [13] D. Holliday, R. Resnick, J. Walker, "Fundamentals of physics," John Wiley and Sons, 1993.
- [14] E. Rashedi, "Gravitational Search Algorithm," M.Sc. Thesis, Shahid Bahonar University of Kerman, Kerman, Iran, 2007.
- [15] J Dong, H Wang, Y Li and S Cheng "Virtual machine placement optimizing to improve network performance in cloud data centers," The Journal of China Universities of Posts and Telecommunications, Vol. 21, No. 3, pp. 62-70, 2014.
- [16] M. Cheng, D. Prayogo, "Symbiotic Organisms Search: A new metaheuristic optimization algorithm," Computers & Structures, Vol. 139, No. 15, pp. 98-112, 2014.
- [17] M. Eusuff, K. Lansey, F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization," Civil Engineering and Engineering Mechanics, Vol. 38, No. 2, pp. 129-154, 2006.
- [18] A. Marphatia, A. Nuhnot, T. Sachdeva, E. Shukla, L. Kurup, "Optimization of FCFS Based Resource Provisioning Algorithm for Cloud Computing," IOSR Journal of Computer Engineering (IOSR-JCE), Vol. 10, No. 5, pp. 01-05, 2013.
- [19] B. Santosa, L. Safitri, "Biogeography-based Optimization (BBO) Algorithm for Single Machine Total Weighted Tardiness Problem (SMTWTP)," The International Journal of Advanced Manufacturing Technology, Vol. 58, No. 9, pp. 1115-1129, 2011.