بهبود تخصیص منابع در محاسبات ابری با استفاده از روش نیمه مارکوف

زهرا توکلی*[1]، محمدرضا خیام‌باشی[2]

۱- مؤسسه غیرانتفاعی صفاهان اصفهان، **z.tavakoli@safahan.ac.ir**

۲- دانشیار دانشکده مهندسی کامپیوتر دانشگاه اصفهان

چکیده

رایانش ابری[1] بر پایه‌ی شبکه‌های رایانه‌ای مانند اینترنت است که الگویی تازه برای عرضه، مصرف و تحویل خدمات رایانشی با به کارگیری شبکه ارائه می‌کند. یکی از مشکلات محاسبات ابری مربوط به بهینه‌سازی تخصیص منابع است. تخصیص منابع با هدف به حداقل رساندن هزینه‌ها، زمان و موارد دیگر انجام می‌گیرد. یکی از چالش‌های تخصیص منابع، می‌توان رسیدگی به تقاضای روزافزون مشتریان و الزامات برنامه‌های کاربردی نام برد. هر چه خدمات مبتنی بر ابر بیشتر شود، نیاز به تأمین منابع بیشتر و چالش مسئله تخصیص منبع، بیشتر می‌شود. در روش‌های ساده و مستقیم، یا همه‌ی منابع در دسترس در مراکز داده به هر درخواست سرویس اختصاص داده می‌شود (سیاستAU)[2]. یا یک مقدار ثابت از منابع را به هر درخواست سرویس اختصاص می‌دهند (سیاست ثابت).

در این پژوهش، تخصیص منابع با روش نیمه‌مارکوف با در نظر گرفتن زمان و هزینه سرویس بررسی شده‌است. در مدل پیشنهادی این پژوهش، روش تصمیم‌گیری نیمه مارکوف[3] تنها یک بخش از منابع را برای هر درخواست رزرو می‌کند. این روش، میانگین سود کلی را بهبود می‌بخشد و می‌تواند احتمال رد درخواست‌های سرویس را کاهش دهد. همچنین از بیشترین منابع پیش خریداری شده استفاده می‌کند و درنتیجه می‌تواند میانگین هزینه پیشنهاد شده برای کاربر در مقایسه مدل پرداخت در هر بار استفاده کاهش دهد. جهت ارزیابی روش پیشنهادی این پژوهش، رابطه نرخ تولید سرویس با کارایی، دقت سرویس، سود منابع ایستا و سود منابع مورد تقاضا مورد بررسی قرار گرفته است. نتایج حاکی از آن است که سود به واقعیت نزدیک می‌کند، هزینه را کاهش می‌دهد، کارایی و دقت‌سرویس افزایش می‌یابد.

کلمات کلیدی: تخصیص منابع ، رایانش ابری، روش نیمه‌مارکوف، منابع مورد تقاضا

[1] Cloud Computing
[2] All Utility Policy (AU)
[3] Semi Markov Decision Process(SMDP)

۱– مقدمه

یک ابر، یک نوع سیستم توزیع‌شده و موازی است که شامل مجموعه‌ای از کامپیوترهای به هم‌پیوسته و مجازی می‌باشد که به طور پویا یک یا چند منبع رایانش ابری را برحسب توافق‌نامه[1] سطح سرویس تأمین می‌کند. این خدمات از طریق مذاکرات بین ارائه دهنده و مصرف‌کنندگان ارائه می‌شوند [۱].

به طور کلی ابرها را می‌توان به سه دسته تقسیم کرد: زیرساختار به عنوان سرویس[2]، پلت‌فرم به عنوان سرویس[3] و نرم‌افزار به عنوان سرویس[4]. برای تخصیص منبع در رایانش ابری، برای هر سرور، تعداد ماشین‌های مجازی و توان‌های آن‌ها محاسبه می‌شوند و پهنای باند شبکه، خط کیفیت[5]، زمان پاسخ، هزینه وظایف، قابلیت اطمینان و موارد دیگر هم تحلیل می‌شود [۲].

مجازی‌سازی، یک راه حل مؤثر برای مدیریت پویای منابع در مراکز داده ابری است. مهم‌ترین جنبه تکنولوژی مجازی‌سازی، تخصیص منابع فیزیکی به سرورهای مجازی است. این تکنولوژی به مراکز داده اجازه می‌دهد که به طور همزمان، چندین ماشین مجازی[6] روی تنها یک دستگاه فیزیکی اجرا شوند و تنظیم ظرفیت منبع (عمدتاً ریزپردازنده و حافظه) را تضمین می‌کند [۳].

مساله مدیریت منبع شامل تخصیص، فراهم‌کردن، نگاشت نیازمندی، بازیابی، انطباق، مبادله، تخمین و مدلسازی و ... می‌باشد. مزایای مدیریت منبع در محاسبات ابری همچون: مقیاس‌پذیری، کیفیت سرویس، سود بهینه، کاهش سربار، بهبود خروجی، کاهش تأخیر، محیط تخصصی، بهبود هزینه و ساده‌سازی واسط کاربری را فراهم می‌کند [۴].

رویکردی که در سرویس‌های محاسبات ابری دنبال می‌شود شامل پرداخت مبلغ مشخصی برای منابع مورد تقاضا و در اختیار گرفتن منبع برای مدت زمان توافق شده‌است. در حال حاضر از این روش، سرویس‌دهنده‌های معروف ابری از جمله شرکت‌های EC2 آمازون استفاده می‌کنند. مبلغ به ازاء استفاده ساعتی از ریزپردازنده محاسبه می‌شود. اگر چه این روش از روش‌های ابتدایی ارائه کالا یا خدمات می‌باشد، اما از تعامل پیچیده بین خریدار و سرویس‌دهنده ابری جلوگیری می‌کند. یکی از ایرادات این روش این است که در زمان بالابودن بارکاری سیستم، یک کار که توسط یک مشتری تقاضا داده می‌شود، می‌بایستی مدت زمان زیادی منتظر بماند تا بار کاری کمتر شده و به کار مربوطه، منابع مورد نیاز اختصاص داده شود. روش‌های فعلی اختصاص منابع از قبیل نوبت گردشی[7] و اولین ورودی، اولین خروجی[8] می‌باشد که بدون در نظر گرفتن اولویت کاری بین کارها نوعی تخصیص غیر عادلانه را انجام می‌دهند. مشتریان علاقه‌مند هستند که کارهایشان در کمترین زمان ممکن و با کمترین هزینه به اتمام برسانند. از طرف دیگر، سرویس‌دهنده ابر نیز تمایل دارد میزان استفاده از منابع خود را به حداکثر برساند. همچنین میزان سود خود را افزایش دهند که این دو در تضاد با یکدیگر هستند که معمولاً درنظر گرفتن همه‌ی این موارد با روشهای سنتی اختصاص منبع و مکانیزم‌های زمان بندی موجود امکان‌پذیر نیست [۵].

در تخصیص منبع می‌توان فرض کرد که به طور کلی حالت فعلی سیستم وابسته به حالت قبلی یا به حالات قبل‌تر می‌باشد. بنابراین می‌توان از روش‌های مارکوف استفاده کرد. به طور گسترده روش‌های مارکوف با مسائل مدیریت منبع سازگار هستند.

---

[1] Service Level Argeement(SLA)
[2] Infrastructure as a Service(IaaS)
[3] Platform as a Service(PaaS)
[4] Software as a Service(SaaS)
[5] Quality Line
[6] Virtual Machine(VM)
[7] Round Robin
[8] First in ,First out

در روش مارکوف گسسته، برای توصیف مناسب یک سیستم فعلی، نیاز به دانستن حالت فعلی در کنار تمام حالات قبلی است. در زنجیره مارکوف مرتبه اول، توصیف احتمالی تنها با حالت فعلی و حالت قبلی معین می‌شود. سپس مدل مارکوف مرتبه nام، توصیف احتمالات سیستم با حالت فعلی و n حالت قبلی مشخص می‌شود. همچنین مدل مارکوف مخفی، با توجه به دانسته‌های قبلی و یک سری داده اولیه یا حالت آغازین، به یک حالت یا حالت پایانی می‌رسد. در این مدل، توابع احتمالاتی از مشاهدات حالت بدست می‌آید [۶].

مدل نیمه مارکوف تا حدودی شبیه مدل مارکوف است و مفاهیم نظیر آن از قبیل وضعیت فعلی، گذار از وضعیتی به وضعیت دیگر را شامل می‌شود. ولی زنجیره‌های نیمه‌مارکوفی در بسیاری از پدیده‌های واقعی موفق‌تر از بکارگیری زنجیره‌های مارکوف عمل می‌کنند. روش نیمه‌مارکوف به نوعی فضای بین زنجیره‌های مارکوف زمان گسسته و زمان پیوسته را پوشش می‌دهد و نسبت به فرآیند مارکوف محدودیت کمتری دارد. در روش تصمیم‌گیری نیمه‌مارکوف تنها یک بخش از منابع را برای هر درخواست رزرو می‌کند. این روش، می‌تواند احتمال رد درخواست‌های سرویس را کاهش دهد و همچنین میانگین سود کلی را بهبود بخشد [۷].

درنتیجه می‌توان گفت روش نیمه‌مارکوف می‌تواند گزینه مناسب برای مدلسازی تخصیص منابع در محیط ابری باشد. بنابراین هدف ما، ارائه یک مدل مناسب برای بهبود تخصیص کارآمد منبع و توانایی تخصیص پویای درخواست‌های سرویس در محیط ابری، با استفاده از یکی از روش‌های مارکوف مانند روش نیمه مارکوف می‌باشد. که از یک طرف این مدل بتواند، با درنظر گرفتن هزینه‌های متحمل شده و زمان سرویس، سود سیستم را حداکثر بسازد و همچنین با توجه به افزایش درخواست کاربران و کمبود فضا در حافظه‌های فیزیکی، این مدل بتواند از عدم رضایت کاربر و افزایش احتمال رد درخواست کاربران جلوگیری نماید.

۲– کارهای گذشته

تاکنون روش‌های مختلفی برای اختصاص منبع ارائه شده از جمله می‌توان به روش‌های مارکوف، مجازی‌سازی، MCDM، TOPSIS، SMDP، GRMP-Q، AHP و دیگر روش‌ها می‌توان اشاره کرد.

شیائو و همکارانش[۸]، سیستمی را با استفاده از تکنولوژی‌های مجازی‌سازی برای اختصاص منابع به صورت پویا در مراکز داده ارائه داده‌اند که براساس تقاضاهای کاربردی و پشتیبانی از رایانش ابری به بهینه‌سازی تعدادی از سرورها می‌پردازند. از معیار TPC-W، که یک معیار استاندارد صنعتی برای برنامه‌های تجارت الکترونیکی می‌باشد، استفاده شده‌است. این معیار مرتبط با انواع حجم کاری همچون مرور کردن صفحات وب، خرید کردن، حجم کاری ابر متن و موارد دیگر است.

مگیوری، یک مدل تصادفی از محاسبات ابری را بررسی می‌کند که کارها بر طبق یک فرآیند تصادفی وارد می‌شوند. در این مدل، مساله تخصیص منبع را از مساله تعدیل کردن بار جدا کرده است. کاربران درخواست منابعشان را در فرم ماشین‌های مجازی می‌فرستند. در هر درخواست تعدادی منابع مانند: پردازنده، حافظه، فضای ذخیره‌سازی تقاضا می‌شود. فراهم‌کننده‌های سرویس، ابتدا درخواستها را صف‌بندی می‌کنند و سپس برای اجرا روی ماشینهای فیزیکی (سرورها) برنامه‌ریزی می‌کنند. هر سرور به تعدادی منابع از هر نوع، محدود شده است[۹].

اُدی و همکاران [۱۰]، تخصیص منابع چند ابری با استفاده از روش مارکوف مطرح نموده‌اند و از یک واسطه مدیریت ابری برای بهینه‌سازی تخصیص منبع و رضایتمندی کاربران استفاده کرده‌اند. در این روش، سود منابع حداکثر شده‌است و به سود بالاتری نسبت به روش حریصانه دست پیدا کرده‌است.

فنگ و همکارانش[۱۱]، برای متعادل کردن بار توزیع شده ماشین‌های مجازی در مرکز داده ابری از روش TOPSIS استفاده کرده‌اند. سیستم می‌تواند تعادل بار بهتری را در یک مقیاس بزرگ با مهاجرت کمتر VMها در محیط رایانش ابری فراهم کند. روش TOPSIS، یکی از کارآمدترین تکنیک‌های تصمیم‌گیری چند معیاره است. این روش می‌تواند مناسب‌ترین ماشین فیزیکی در مرکز داده را برای مهاجرت کردن VMها پیدا کند.

وهیب وهمکارانش، طرحی جدید در راستای پیاده‌سازی و ارزیابی مدیریت منابع سیستم روی Open Stack را ارائه می‌دهند. یک Open Stack می‌تواند سطحی برای ابرهای عمومی و خصوصی باشد. Gossip، یک پروتکل عمومی برای مدیریت خاص منبع در محیط‌های ابری می‌باشد. همچنین پروتکل Gossip، کنترلگر جایگزینی پویا را پیاده‌سازی می‌کند [۱۲].

هدف وانگ و همکارانش[۱۳]، حداکثرسازی سود کلی می‌باشند. هزینه کلی برای خدمت‌رسانی مشتری وابسته به میانگین زمان پاسخ درخواست برای هر مشتری که توسط تابع کارایی تعریف شده‌است، محاسبه می‌شود. وانگ، یک راه حل سلسله مراتبی نزدیک به بهینه را پیشنهاد می‌کند. که شامل یک مدیر مرکزی و عامل‌های محلی توزیع شده می‌باشد. مدیر مرکزی، مسئله گسیل درخواست و پیداکردن تعداد بهینه سرورهای فعال برای پردازش درخواست‌ها را حل می‌کند. سپس یک تبادل مطلوب بین زمان پاسخ درخواست سرویس و مصرف توان سیستم را بدست می‌آورد.

رائو و همکاران [۱۴] و آلپکان و همکاران [۱۵]، از نظریه بازی‌ها برای مسئله تخصیص منابع استفاده کرده‌اند. این راه‌حل طی دو مرحله زیر ارائه شده‌است. اول، هر شرکت‌کننده مشکل خود را به طور مستقل، بدون در نظر گرفتن تخصیص منابع چندگانه حل می‌کنند. روش برنامه‌نویسی دودویی عدد صحیح برای حل بهینه‌سازی مستقل ارائه شده‌است. دوم، یک مکانیزم تکاملی طراحی شده‌است، که از استراتژی تسهیم منبع با به حداقل رساندن زیان‌های بهره‌وری خود استفاده می‌کند.

آقای گودرزی و همکارانش[۱۶]، مسئله اختصاص منبع را روی سوددهی مطرح کرده‌اند. برای حل این مسئله، یک الگوریتم براساس جست‌وجوی نیروی هدایت‌شده ، پیشنهاد داده‌اند. پردازش، حافظه مورد نیاز و منابع ارتباطی، به عنوان سه بعد در بهینه‌سازی مورد توجه قرار داده‌اند. اغلب تأمین منابع، زیرساختار ارائه‌دهندگان بیشتر به منظور پاسخ‌گویی نیازهای مشتریان براساس SLA می‌باشد. در این‌جا دو نوع کلاس SLA شامل کلاس طلایی توافق‌نامه‌های سطح سرویس و کلاس برنز توافق‌نامه‌های سطح سرویس وجود دارد.

داجی و همکارانش، یک مدل برای تخصیص منبع وظیفه‌گرا در یک محیط رایانش ابری معرفی کرده‌اند. که تخصیص منبع توسط تکنیک ماتریس مقایسه جفتی و فرآیند تحلیل سلسله مراتبی به منابع در دسترس و با توجه به ترجیحات کاربر انجام می‌شود. منابع محاسباتی می‌تواند برحسب وزن وظایف اختصاص داده شود [۱۷].

لی و بیون جونگ ژوئن و همکارانش، با استفاده از طرح تصمیم‌گیری فازی مشکل انتخاب ماشین مجازی و ماشین فیزیکی مناسب برای مهاجرت را حل می‌کنند. که یک روش جدید برای تخصیص منبع ماشین‌های مجازی پیشنهاد کرده‌اند که مفهوم TOPSIS و فرآیند تحلیل سلسله مراتبی، تئوری خاکستری و مفاهیمی از آنتروپی به کار می‌گیرد [۳].

ژن و همکارانش[۱۸]، یک مدل اختصاص منبع برای به کارگیری برنامه‌های کاربردی SaaS روی سطوح رایانش ابری را ارائه کرده‌اند که با استفاده از روش چند اجاره‌ای یک محیط مقیاس‌پذیر و مقرون‌به‌صرفه را ایجاد کرده‌است. در این مدل، تقاضا دسترسی به منابع محاسباتی همراه با مدل پرداخت در هر بار استفاده بررسی‌شده‌است. ارائه‌دهندگان برنامه‌های کاربردی با مقیاس یکپارچه، سرویس‌ها را فراهم می‌کنند.

تاریک و همکاران [۱۹]، مسئله مدیریت منبع در سیستم‌های ابری جغرافیایی توزیع شده را در نظرگرفته‌اند و به دنبال مفهوم ابر Me که امکان سرویس مهاجرت را، در میان مراکز داده‌ای فراهم کند، می‌باشند. وقتی موبایل کاربران درمناطق سرویس حرکت می‌کند، دو نوع درخواست سرویس به مراکز داده وجود دارد. یعنی درخواست‌های جدید[1] در منطقه

سرویس محلی شروع شده و یا درخواست‌های مهاجرت[۲] صادر شده‌است. بنابراین استفاده از روش مارکوف برای کمک به مدیر منبع پیشنهاد شده‌است که تصمیم بگیرند آیا درخواست‌های سرویس را قبول کنند یا نه. اگر درخواست پذیرفته شود چه مقدار منابع باید به هر سرویس تخصیص داده شود. در این‌جا هدف، بهبود سود سیستم کلی می‌باشد.

۳- روش پیشنهادی

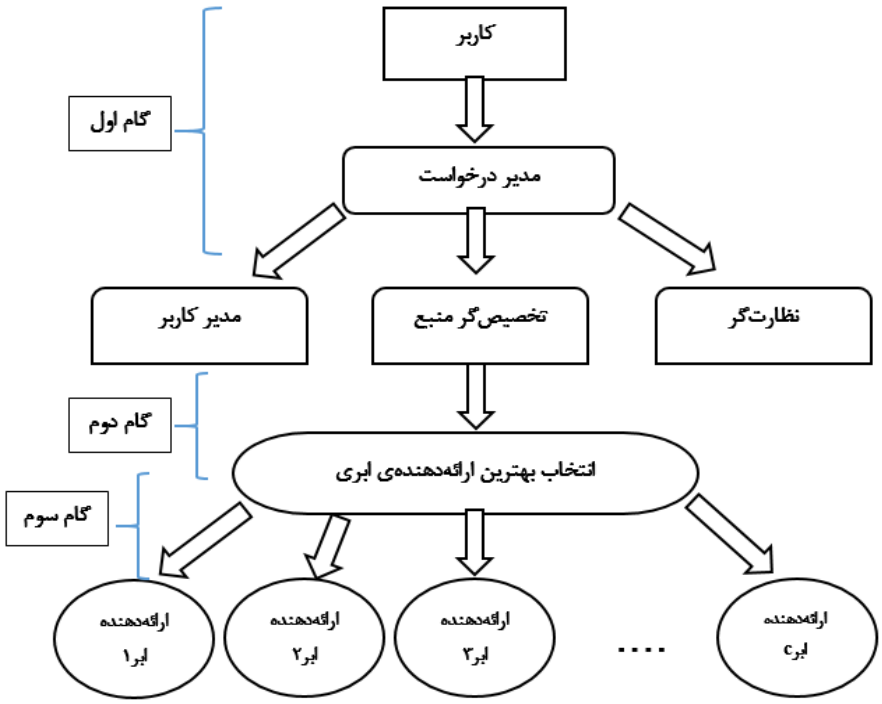
در این پژوهش، فرآیند مورد نظر در سه گام انجام می‌شود.

۱- ارسال درخواست کاربر به تخصیص‌گر منبع

۲- انتخاب ارائه‌دهنده ابری مناسب با استفاده تابع ارزش حالت در الگوریتم تکرار ارزش برای سرویس‌دهی به درخواست وارد شده توسط تخصیص‌گر منبع

۳- ارسال درخواست به ارائه‌دهنده انتخاب شده و اجرای آن درخواست توسط آن ارائه‌دهنده

به طور کلی فرآیند دریافت درخواست از کاربر تا ارسال به ارائه‌دهنده مناسب در شکل۱، تشریح شده‌است.



شکل ۱، فرآیند دریافت درخواست از کاربر تا ارسال به ارائه‌دهنده مناسب

کاربر، درخواست به مدیر درخواست ارسال می‌کند. سپس مدیر درخواست، درخواست را دریافت می‌کند و آن را به تخصیص‌گر منبع تحویل می‌دهد، تخصیص‌گر منبع، نقش اصلی را در این روش پیشنهادی بازی می‌کند. این مؤلفه دارای الگوریتم تکرار ارزش روش نیمه مارکوف است که با استفاده از آن وقتی سیستم در حالت i ام است، بهترین حالت مطلوب بعدی (حالت j ام) را با توجه این‌که سود سیستم حداکثر شود، انتخاب می‌کند. وقتی بهترین ارائه‌دهنده انتخاب شد، درخواست به آن ارائه‌دهنده برای اجرا تحویل داده می‌شود. مؤلفه‌ی نظارت‌گر، بر

وضعیت ارائه‌دهنده نظارت می‌نماید و پارامترهای دقت‌سرویس[1]، تعداد رد درخواست‌ها و کارایی را بعد از انجام سرویس محاسبه می‌کند.

در مؤلفه‌ی تخصیص‌گر منبع، تخصیص منابع به دو صورت زیر انجام می‌گردد.

۱) براساس تقاضا، به معنی استفاده از یک مدل هزینه، پرداخت در هربار استفاده (منابع مورد تقاضا)

۲) استفاده از یک مجموعه منابع از پیش خریداري شده و متعلق به ارائه دهندگان ابري (منابع ایستا)

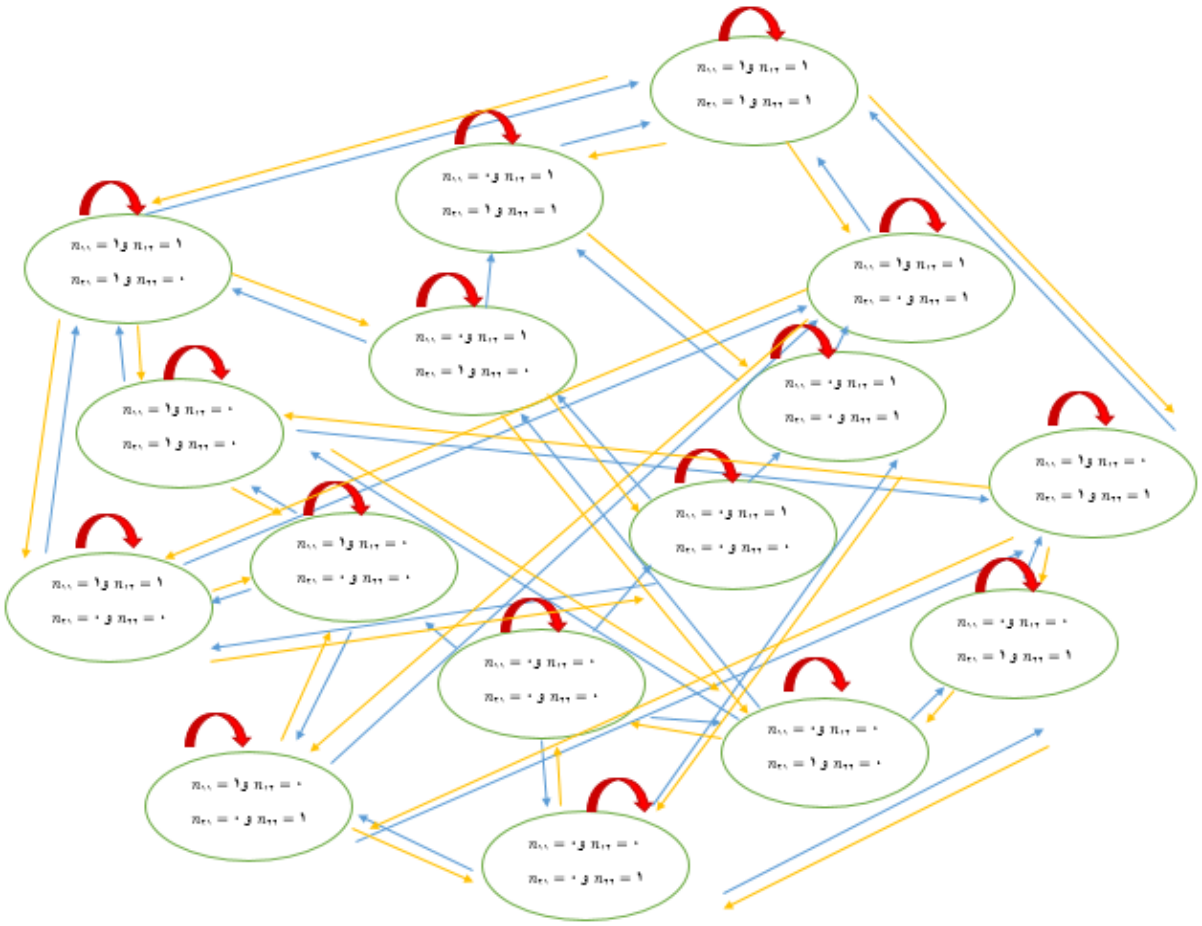
اگر منابع ایستا در ابرها موجود باشد، تابع تکرار ارزش مشخص می‌کند این درخواست وارد شده برای سرویس‌دهی به کدام ارائه‌دهنده ابر ارسال شود. که سود سیستم را با توجه به زمان و هزینه سرویس افزایش یابد. اگر منابع ایستا در ابرها وجود نداشته باشد. درخواست‌ها در صف قرار می‌گیرند که منابع خریداری شود و به نوبت در اختیار این درخواست‌ها گذاشته شود.

فرآیند تصمیم نیمه‌مارکوف روش پیشنهادی ما در زمان پیوسته با چند متغیر تصادفی به صورت زیر تعریف می‌شود:

$\{S, A, T, r, t, co\}$

S، مجموعه حالت‌ها را نشان می‌دهد که مقدار منابع ایستا ارائه‌دهنده‌ها، تعداد ارائه‌دهنده‌ها و تعداد نوع سرویس‌ها در تعیین اندازه این مجموعه حالات دخیل می‌باشد. A مجموعه اقدامات ممکن در حالات، T ماتریس احتمال انتقال، r ماتریس سود، co ماتریس هزینه سرویس و t ماتریس زمان سرویس می‌باشند. اعمال a^{o}_{ck} و a^{v}_{ck} ، به ترتیب عمل تخصیص سرویس از نوع k به منابع ایستا روی ابر c و عمل تخصیص سرویس از نوع k به منابع مورد تقاضا روی ابر c را نشان می‌دهند. برای هر ارائه‌دهنده ابر، منابع ثابتی وجود دارد. این منابع از پیش خریداری شده برای هر ابر c، با W_c نشان داده می‌شود. مقدار منابع مورد نیاز برای سرویس‌دهی هر نوع سرویس k، با W_k نشان داده می‌شود.

[1] Service Accuracy (SA)



شکل۲، نمونه حالات برای دو تا ارائه‌دهنده و دو نوع سرویس

در شکل۲، یک نمونه با ۱۶ تا حالت مشاهده می‌شود. در اینجا n_{ck} تعداد سرویس‌های موجود از نوع k روی ابرc را مشخص می‌کند. فلش‌های با رنگ آبی نشان می‌دهد که یک درخواست جدید به منابع ایستا تخصیص داده شده است و فلش‌های خود انتقالی با رنگ قرمز نشان می‌دهد که درخواست جدید به منابع مورد تقاضا تخصیص داده شده است. سپس فلش‌های نارنجی به اتمام رسیدن سرویس‌دهی آن درخواست منابع ایستا را نمایش می‌دهد. بنابراین برای دو تا ارائه‌دهنده و یک نوع سرویس، ۴ تا حالت وجود دارد و برای دوتا ارائه‌دهنده و دو نوع سرویس، ۱۶ تا حالت وجود دارد.

۱-۳- پارامترهای مورد استفاده در روش پیشنهادی

پارامترهای مورد استفاده در این روش پیشنهادی عبارتند از:
زمان سرویس: از زمانی که سرویس، منبع مورد نظر خود را در اختیار می‌گیرد و تا زمانی که سرویس انجام شده و منبع آزاد

می‌شود، زمان سرویس گفته می‌شود. t(i,a,j)، زمان سرویس از حالت iام به حالت jام تحت تأثیر عمل a می‌باشد.

هزینه سرویس: این هزینه می‌تواند هزینه نگهداری فضای مورد نظر برای درخواست یا هزینه مصرف انرژی باشد. $co(i,a,j)$ هزینه سرویس از حالت iام به حالت jام تحت تأثیر عمل a، توسط معادله۱، محاسبه می‌شود:

$$co(i,a,j) = c(i,a,j)t(i,a,j) \quad (1)$$

$c(i,a,j)$ نرخ هزینه می‌باشد که مقدار آن در حالت‌های مختلف متفاوت است. نحوه‌ی محاسبه نرخ هزینه سرویس به صورت زیر است:

۱- وقتی به درخواست جدید منابع ایستا تخصیص داده شود، آنگاه نرخ هزینه براساس تعداد منابعی که ابر c را اشغال می‌کند و منجر به هزینه شده است، محاسبه می‌شود. مجموع منابع اشغال شده در هر ابر c باید کمتر یا مساوی وزن W_C آن ابر باشد $(\sum_{k=1}^{K}(W_k\, n_{ck}) \le W_n)$.

۲- وقتی به درخواست جدید منابع مورد تقاضا تخصیص داده شود، هزینه براساس مدل پرداخت در هر بار استفاده محاسبه می‌شود.

۳- در غیر این صورت، مقدار هزینه صفر می‌شود.

کارایی: همان‌طور که در رابطه۲، مشاهده می‌شود، کارایی هر حالت، از تقسیم زمان اجرا سرویس به کل زمان فراخوانی سرویس می‌باشد. که در روش پیشنهادی، این زمان‌ها و سپس کارایی هر حالت محاسبه می‌شود.

$$\text{کارایی} = \frac{\text{زمان اجرا سرویس}}{\text{کل زمان فراخوانی سرویس}} \quad (۲)$$

دقت سرویس: معیار دقت سرویس، از تعداد سرویس‌های پاسخ داده شده به کل سرویس‌های موجود محاسبه می‌شود، همان‌طور که در رابطه۳، دیده می‌شود.

$$SA = \frac{\text{تعداد سرویس‌های پاسخ داده شده}}{\text{تعداد کل درخواست‌ها}} \quad (۳)$$

هر چه مقدار دقت‌سرویس افزایش یابد، کارایی بیشتر می‌شود. مقدار دقت سرویس بین صفر و یک می‌باشد، وقتی تعداد رد درخواست‌ها به صفر برسد، مقدار دقت سرویس یک می‌شود.

تعداد نوع سرویس‌ها (K): یکی از مزیت‌های روش نیمه‌مارکوف نسبت به روش مارکوف، این است که هر گام زمانی آن می‌تواند هر توزیع دلخواهی با مقادیر مثبت باشد. در حالی که در روش مارکوف هر گام زمانی فقط از یک توزیع خاصی می‌تواند پیروی کند. بنابراین محدودیت‌های روش نیمه‌مارکوف کمتر از روش مارکوف می‌باشد. آنگاه در این پژوهش با استفاده روش نیمه‌مارکوف از نرخ‌های مختلف (ثابت، نمایی و پواسون) برای تولید سرویس استفاده می‌شود.

ماتریس احتمال انتقال و ماتریس سود: نحوه‌ی محاسبه ماتریس احتمال انتقال $T(i,a,j)$ توسط معادله۴، همانند ماتریس احتمال انتقال در روش مارکوف محاسبه می‌شود.

$$T(i,a,j) = \begin{cases} \overline{\lambda_k} a_{ck}^{v} & if\ \acute{s} = s + \delta_{ck}, k\epsilon K, c\epsilon C \\ \overline{\mu_k}\, n_{ck}\ if\ \acute{s} = s - \delta_{ck}, k\epsilon K, c\epsilon C, n_{ck} > 0 \\ 1 - \sum_{c\epsilon C}\sum_{k\epsilon K}(\overline{\lambda_k} a_{ck}^{v} + \overline{\mu_k} n_{ck}) & if\ \acute{s} = s \\ 1, otherwise \end{cases} \quad (۴)$$

زمان انتقال مورد انتظار در حالت i ام با انتخاب عمل a ام به صورت معادله ۵، محاسبه می‌شود.

$$y(i,a) = \sum_{j} T(i,a,j)\, t(i,a,j) \tag{5}$$

عناصر ماتریس انتقال برحسب زمان به صورت معادله ۶، محاسبه می‌شود:

$$T^{t}(i,a,j) = \begin{cases} \dfrac{\eta \overline{\lambda_k} a^{v}_{ck}}{y(i,a)} & if\ j = i + \delta_{ck}, k \epsilon K, c \epsilon C \\ \dfrac{\eta \overline{\mu_k} n_{ck}}{y(i,a)} & if\ j = i - \delta_{ck}, k \epsilon K, c \epsilon C, n_{ck} > 0 \\ 1 - \eta \left[\dfrac{\sum_{c \epsilon C} \sum_{k \epsilon K} \left(\overline{\lambda_k} a^{v}_{ck} + \overline{\mu_k} n_{ck} \right)}{y(i,a)} \right] & if\ j = i \\ 0 & otherwise \end{cases} \tag{6}$$

مقدار η، به صورت زیر می‌باشد:

$$0 \leq \eta \leq \frac{y(i,a)}{1 - T(i,a,j)}$$

ماتریس سود به صورت معادله ۷، محاسبه می‌شود:

$$r(i,a,j) = \begin{cases} \left[\left(r^{v}_{ck} a^{v}_{ck} \right) - \sum_{k=1}^{K} (W_k n_{ck}) t(i,a,j) \right] & if\ j = i + \delta_{ck} \\ \left[\sum_{k \epsilon K} \left(\dfrac{\overline{\lambda_k} a^{v}_{ck}(i)}{T(i,a,j)} \right) r^{v}_{ck} \right] - [c(i,a,j) t(i,a,j)] & if\ j = i \\ 0 & otherwise \end{cases} \tag{7}$$

پارامترهای مؤثر دیگر از جمله: زمان تأخیر صف، طول صف، تعداد رد درخواست‌ها، نوع سرویس، تعداد ارائه‌دهنده‌ها (C)، نرخ تولید سرویس نوع k (λ_k) و نرخ سرویس‌دهی به سرویس نوع k (μ_k) می‌باشند.

۲-۳- الگوریتم تکرار ارزش

r^{v}_{ck} و a^{v}_{ck}، سود پایه‌ای حاصل توسط تخصیص یک نوع سرویس جدید k که به ترتیب توسط منابع ایستا و منابع مورد تقاضا روی ابر c قابل دسترس هستند. این سود می‌تواند برحسب توان مصرفی یا انرژی مصرفی، رضایت کاربر یا اهمیت نوع سرویس یا میزان نزدیکی ارائه‌دهنده به کاربر از نظر مسافت باشد.

مقدار میانگین سود لحظه‌ای مورد انتظار $\bar{r}(i,a)$ در حالت i ام با انتخاب عمل a ام به صورت معادله ۹ محاسبه می‌شود.

$$\bar{r}(i,a) = \sum_{j} T(i,a,j) r(i,a,j) \tag{9}$$

مقادیر میانگین سود لحظه‌ای بر حسب زمان $\bar{r}^{t}(i,a)$ به صورت فرمول ۱۰ محاسبه می‌شود:

$$\bar{r}^{t}(i,a) = \frac{\bar{r}(i,a)}{y(i,a)} \tag{10}$$

تابع ارزش حالت $V^{*}(i)$ در یک حالت i ام که $i \epsilon S$، تحت سیاست π حاصل می‌شود. از این تابع برای ارزیابی مجموعه سیاست‌های ممکن در جهت بهبود تخصیص منابع استفاده می‌شود. سیاست π، با انتخاب اعمالی که مقدار سمت راست معادله ۸ را ماکزیمم کند، تشکیل می‌شود.

$$V^{*}(i) = \max_{a \epsilon A} \left(\bar{r}^{t}(i,a) + \sum_{j \epsilon S} T^{t}(i,a,j) V^{*}(j) \right) \tag{8}$$

الگوریتم زیر، الگوریتم تکرار ارزش برای روش پیشنهادی را نشان می‌دهد. کار این الگوریتم به این گونه است که بین ارائه‌دهنده‌های مختلف، بهترین ارائه‌دهنده را انتخاب می‌کند، آنگاه با توجه به حالت فعلی بهترین حالت بعدی به گونه‌ایی که سود سیستم حداکثر شود، یافت می‌شود.

$$Initialize\ V\ arbitrarity, (eg, V(i) = 0, \forall\ i \epsilon S)$$
$$Repeat$$
$$\Delta \leftarrow 0$$
$$For\ each\ i \epsilon S$$
$$v \rightarrow V(i)$$
$$v(i) = \max_{a \epsilon A}(\bar{r}^{a}(i,a) + \sum_{j \epsilon s}^{'} T^{a}(i,a,j)v(j))$$
$$\Delta \leftarrow max(\Delta, v - V(i))$$
$$until\ \Delta < \varepsilon\ (\varepsilon\ small\ positive\ number)$$
$$Output\ a\ deterministic\ policy\ \pi$$
$$\pi(i) \leftarrow arg\max_{a \epsilon A}(\bar{r}^{a}(i,a) + \sum_{j \epsilon s}^{'} T^{a}(i,a,j)v(j))$$

۴–ارزیابی شبیه‌سازی

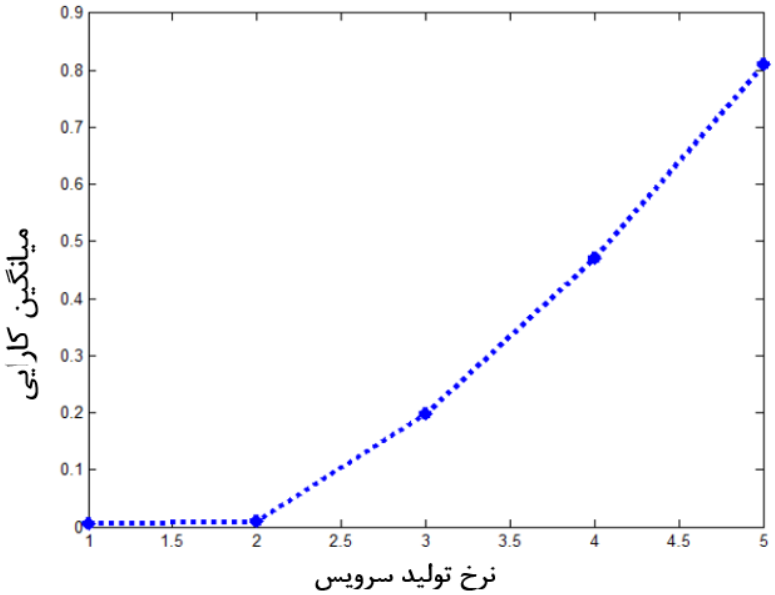
در این پژوهش، برای نرخ ورود و خروج سرویس ها از تابع های توزیع ثابت، پواسون و نمایی استفاده می شود. پیاده سازی روش پیشنهادی در یک شبیه‌ساز رایانش ابری به نام CloudSim صورت گرفته است. این شبیه‌ساز، قابلیت شبیه‌سازی محیط‌های ابری واقعی را تا حد زیادی فراهم کرده‌است. در روش پیشنهادی، معیارهای هزینه، زمان، سود و کارایی مدنظر می باشد. در هیچ کدام از روش های قبلی در محیط ابر، این چهار معیار همزمان در نظر گرفته نشده است. در این شبیه‌سازی به ازای هر موجودیت یک عامل ساخته شده‌است و ارتباط این عامل‌ها با یکدیگر توسط واسط می‌باشد. در این شبیه‌سازی فرض شده‌است که سه تا ارائه‌دهنده و پنج نوع سرویس وجود دارد. جدول ۱، مقادیر پارامترهای اولیه در شبیه‌سازی را نشان می‌دهد.

جدول1، مقادیر پارامترها در ابرها

پارامتر		ابر		
		۱	۲	۳
تعداد ارائه‌دهندگان ابر	C	3		
مقدار منابع ایستا قابل دسترس روی ابر c	**Wc**, c=1,2,3	5	4	4
تعداد سرویس‌ها	K	5		
میانگین فرکانس ورودی برای نوع سرویس k=1,2,3,4,5	(λ_1 , λ_2, λ_3 , λ_4 , λ_5)	(3.51,1.17, 2.1,1.22,2.35)		

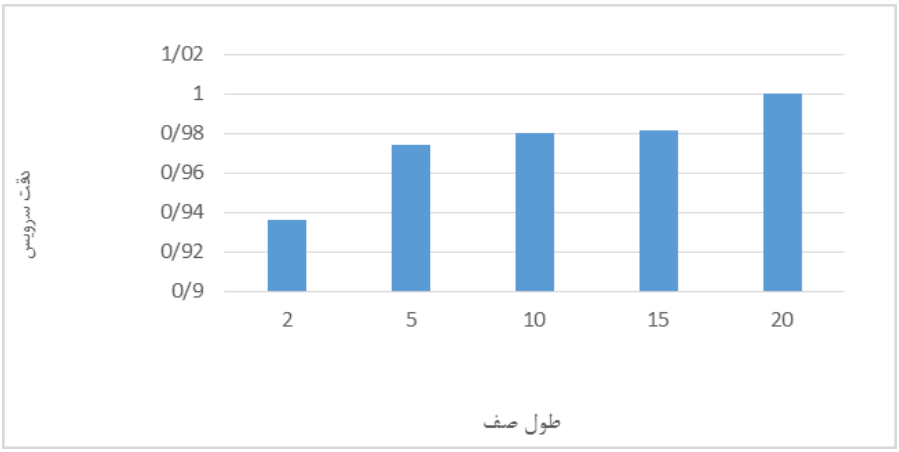
میانگین فرکانس خروجی بـر ا ی نـو ع سـر و یـس k=1,2,3,4,5	$(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5)$	(0.9 , 0.3 , 0.6,0.2,0.4)		
تعداد منابع مورد نیاز	$(w_1, w_2, w_3, w_4, w_5)$	(1,2,3,2,1)		
R^p_{ck} سود حاصل توسط یک سرویس k=1,2,3,4,5 به منابع ایستای قابل دسترس روی ابر c	R^p_{c1} , c=1,2,3	1	0.9	1.25
	R^p_{c2} , c=1,2,3	4	-	5
	R^p_{c3} , c=1,2,3	4	0.9	5
	R^p_{c4} , c=1,2,3	1	0.9	1.25
	R^p_{c5} , c=1,2,3	4	0.9	5
R^o_{ck} سود حاصل توسط یک سرویس k=1,2,3,4,5 به منابع مورد تقاضای قابل دسترس روی ابر c	R^o_{c1} , c=1,2,3	0.2	0.3	0.25
	R^o_{c2} , c=1,2,3	0.35	۰,۴	0.4
	R^o_{c3} , c=1,2,3	0.35	0	0.4
	R^o_{c4} , c=1,2,3	0.2	0.3	0.25
	R^o_{c5} , c=1,2,3	0.35	0	0.4
زمان سرویس		(∞ و ۰,۰۵]		
نرخ هزینه سرویس		(∞ و ۰,۰۵]		
اندازه صف		10		

در شکل ۳، با افزایش نرخ تولید سرویس، میانگین کارایی سیستم افزایش می‌یابد.



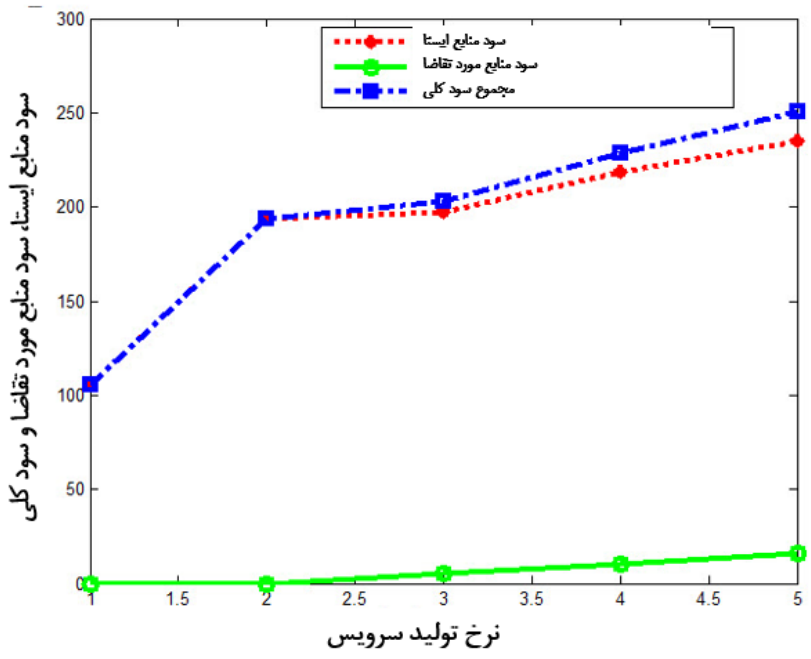
شکل ۳، رابطه نرخ تولید سرویس با میانگین کارایی سیستم

در شکل ۴، با افزایش طول صف درخواست‌های منابع مورد تقاضا، تعداد رد درخواست‌ها کاهش می‌یابد، آنگاه دقت سرویس افزایش پیدا می‌کند. مقدار دقت سرویس بین صفر و یک می‌باشد، وقتی تعداد رد درخواست‌ها به صفر برسد، مقدار دقت سرویس یک می‌شود.



شکل ۴، رابطه طول صف و دقت سرویس در شبیه‌سازی دو تا ارائه‌دهنده ابر و دو نوع سرویس

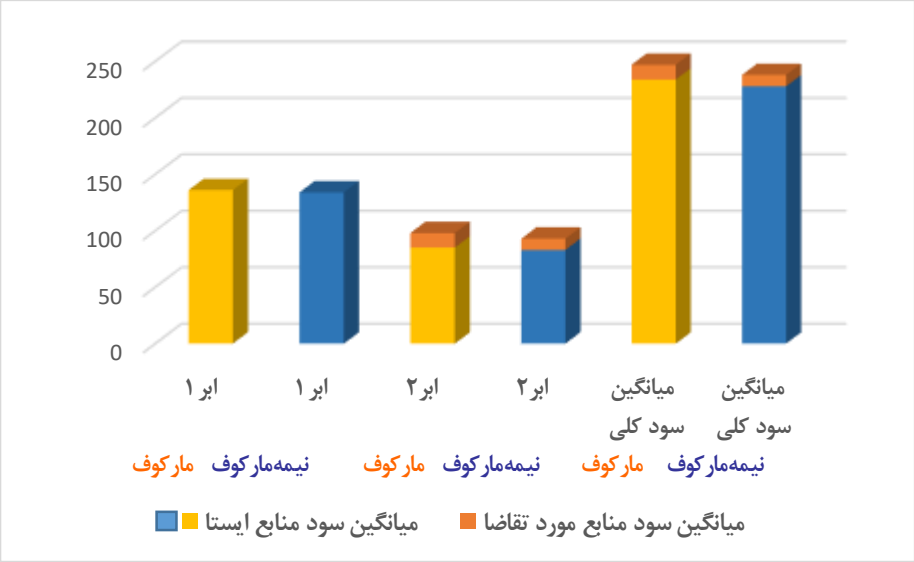
در شکل۵، با افزایش نرخ تولید سرویس، تعداد سرویس‌ها تولیدی بیشتر می‌شود. آنگاه سود منابع ایستا، سود منابع مورد تقاضا افزایش می یابد. شیب نمودار سود منابع ایستا از سود منابع مورد تقاضا بیشتر می‌باشد. از منابع ایستا، نسبت به منابع مورد تقاضا بیشتر استفاده شده‌است.



شکل ۵، نمایش رابطه نرخ تولید سرویس با سود منابع ایستا، سود منابع مورد تقاضا و مجموع سود کلی

در شکل ۶، سود منابع ایستا و سود منابع مورد تقاضا در ابرهای مختلف در روش‌های مارکوف و نیمه مارکوف را نشان می‌دهد. که سود حاصل روش نیمه‌مارکوف مقداری کمتر از روش مارکوف است چون مقدار سود به واقعیت نزدیک‌تر

می‌باشد. در روش نیمه‌مارکوف هزینه و زمان سرویس را مدنظر گرفته است. همچنین برای منابع مورد تقاضا، صف قرار داده می‌شود. در صورتی که منابع ایستا مصرف شده باشد، باید درخواست‌ها را به منابع مورد تقاضا تخصیص داده شود که این درخواست‌ها در صف قرار می‌گیرند تا نوبت سرویس‌دهی آن‌ها برسد. در حالی که روش مارکوف این موارد را در نظر نگرفته‌است.



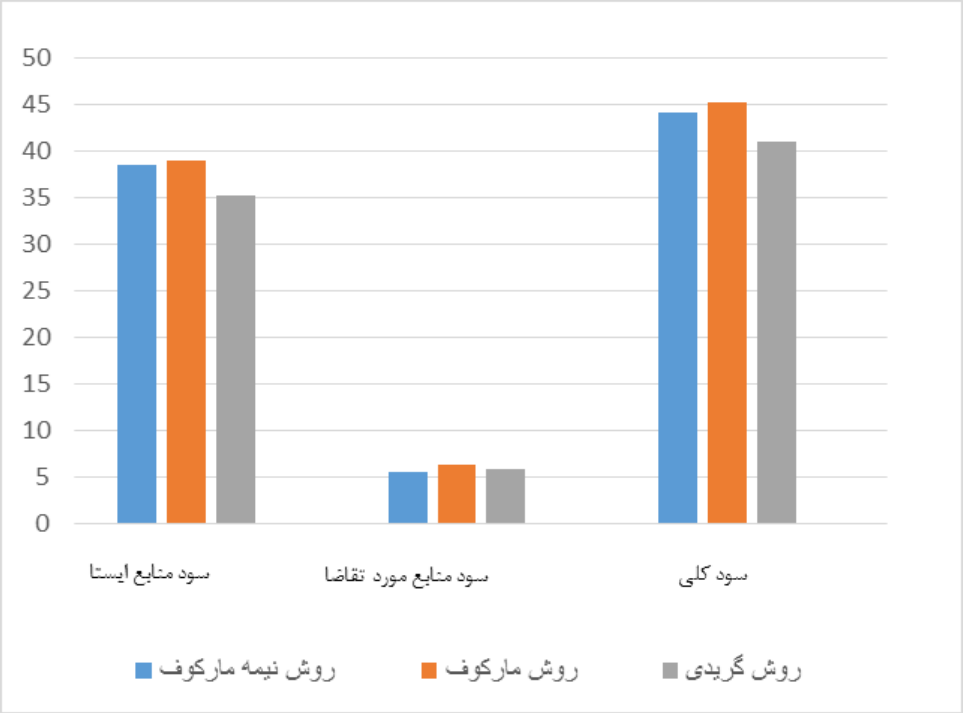
شکل ۶، نمایش سود در ابرها در روش‌های مارکوف و نیمه مارکوف در شبیه‌سازی دو تا ارائه‌دهنده ابر و دو نوع سرویس

در جدول۲ و شکل۷، پارامترهای میانگین سود منابع ایستا، میانگین سود منابع مورد تقاضا و میانگین سود کلی در سه روش گریدی، مارکوف و نیمه‌مارکوف مقایسه می‌کند. که در روش نیمه‌مارکوف نسبت تخصیص منابع مورد تقاضا به تخصیص کل منابع آن کمتر از این نسبت در روش مارکوف و گریدی است و همچنین در این روش پیشنهادی، نسبت تخصیص منابع ایستا به تخصیص کل منابع آن بیشتر از این نسبت در روش مارکوف و گریدی است. بنابراین در روش نیمه‌مارکوف، تخصیص منابع ایستا بیشتر و تخصیص منابع مورد تقاضا کمتر از این تخصیص‌ها در دو روش مارکوف و گریدی است. همچنین میانگین هزینه هم کاهش می‌یابد. در روش نیمه‌مارکوف، معیارهای زمان، هزینه، سود، دقت‌سرویس و کارایی مدنظر قرار داده‌ایم، در حالی که در روش مارکوف فقط معیار سود مدنظر قرار داده شده‌است.

پارامترها	روش گریدی	روش مارکوف	روش نیمه مارکوف
میانگین سود منابع ایستا	35.21	39	38.52
میانگین سود منابع مورد تقاضا	5.85	6.3	5.58

میانگین سود کلی	41.06	45.2	۴4.1

جدول ۲، مقایسه پارامتر سود در سه روش گریدی، مارکوف و نیمه مارکوف



شکل ۷، مقایسه میانگین سود منابع ایستا، مورد تقاضا و میانگین سود کلی

۵–نتیجه‌گیری و پیشنهادات

روش نیمه مارکوف می‌تواند گزینه مطلوبی برای بهبود تخصیص منابع در محیط ابری باشد. تا حدودی توانست با در نظر گرفتن هزینه و زمان سرویس، سود سیستم به واقعیت نزدیک‌تر کند، اگر چه سود حاصل از روش نیمه‌مارکوف کمتر از روش مارکوف شده‌است. همچنین با توجه به افزایش درخواست کاربران و کمبود فضا در حافظه‌های فیزیکی، این مدل می‌تواند از عدم رضایت کاربر و افزایش احتمال رد درخواست کاربران جلوگیری نماید. نسبت تخصیص منابع مورد تقاضا در روش نیمه‌مارکوف کمتر از روش مارکوف می‌باشد این موجب می‌شود که میانگین هزینه پیشنهاد شده برای کاربر در مقایسه مدل پرداخت در هر بار استفاده، کاهش پیدا کند. وقتی منابع ایستا در ابرها تمام شود به درخواست‌ها باید منابع مورد تقاضا تخصیص داده شود. درخواست‌های منابع مورد تقاضا در صف قرار می‌گیرند و به نوبت سرویس‌دهی می‌شوند. وقتی تعداد درخواست‌های مورد تقاضا بیشتر از اندازه صف باشد، سپس آن درخواست رد می‌شود. هر چه طول صف بیشتر ،

تعداد رد درخواست‌ها کمتر، دقت سرویس و کارایی بیشتر می‌شود. در نتیجه این روش پیشنهادی احتمال رد درخواست‌ها را کاهش داده و مقادیر سود واقعی، کارایی و دقت‌سرویس افزایش می‌دهد.

از جمله کارهای آینده که می‌توان در راستا روش پیشنهادی انجام داد. ترکیب ذخیره‌سازی، پردازش و منابع شبکه‌ها برای فراهم کردن خدمات غنی برای کاربران ابر می‌باشد و می‌توان از تقریب‌های روش‌های برنامه‌ریزی پویا در روش MDPها استفاده کرد.

۶- مراجع

1. Ergu, D., Kou, G., Peng, Y., Shi, Y. and Shi, Y.(2013), "The analytic hierarchy process: task scheduling and resource allocation in cloud computing environment". The Journal of Supercomputing, pp.1-14.

2. Ergu D, Kou G, Peng Y, Shi Y.(2011), "The analytic hierarchy process: task scheduling and resource allocation in cloud computing environment. "The Journal of Supercomputing. 1-14.

3. Lee, B., Oh, K.H., Park, H.J., Kim, U.M.(2014), and Youn, H.Y. "Resource Reallocation of Virtual Machine in Cloud Computing with MCDM Algorithm. "In Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), International Conference on (pp. 470-477).

4. Manvi, S.S. and Shyam, G.K. (2014), "Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. "*Journal of Network and Computer Applications*, *41*, pp.424-440.

5. Rao, Nageswara SV, Stephen W. Poole, Fei He, Jun Zhuang, Chris YT Ma, and David KY Yau.(2012), "Cloud computing infrastructure robustness: A game theory approach." In Computing, Networking and Communications (ICNC), 2012 International Conference on, pp. 34-38.

6. Hsu D, Kakade SM, Zhang T. (2012), "A spectral algorithm for learning hidden Markov models. "Journal of Computer and System Sciences. 78(5):1460-80.

7. Chen, J., Long, H., Zheng, Q., Xing, M. and Wang, W. (2015), "An SMDP-based Resource Management Scheme for Distributed Cloud Systems. "In Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st (pp. 1-5).

8. Xiao, Z., Song, W. and Chen, Q. (2013), "Dynamic resource allocation using virtual machines for cloud computing environment. "IEEE transactions on parallel and distributed systems, *24*(6), pp.1107-1117.

9. Maguluri, S.T., Srikant, R. and Ying, L. (2014), "Heavy traffic optimal resource allocation algorithms for cloud computing clusters. "Performance Evaluation, 81, pp.20-39.

10. Oddi, G., Panfili, M., Pietrabissa, A., Zuccaro, L. and Suraci, V., December. (2013), "A resource allocation algorithm of multi-cloud resources based on Markov Decision

Process. "In Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on (Vol. 1, pp. 130-135).

11. Ma, F., Liu, F. and Liu, Z. (2012),"Distributed load balancing allocation of virtual machine in cloud data center." Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on. pp. 20-23.

12. Wuhib, F., Stadler, R. and Lindgren, H. (2012), "Dynamic resource allocation with management objectives—Implementation for an OpenStack cloud." Network and service management (cnsm), 2012 8th international conference and 2012 workshop on systems virtualiztion management (svm). IEEE, pp. 309-315.

13. Wang, Y., Chen, S., Goudarzi, H. and Pedram, M. (2013), "Resource allocation and consolidation in a multi-core server cluster using a Markov decision process model." In Quality Electronic Design (ISQED), 2013 14th International Symposium on, pp. 635-642.

14. Rao, N.S., Poole, S.W., He, F., Zhuang, J., Ma, C.Y. and Yau, D.K. (2012), "Cloud computing infrastructure robustness: A game theory approach. "In Computing, Networking and Communications (ICNC), 2012 International Conference on (pp. 34-38).

15. Alpcan T, Başar T. (2010), "Network security: A decision and game-theoretic approach." Cambridge University Press.

16. Goudarzi, H. and Pedram, M. (2011), "Multi-dimensional SLA-based resource allocation for multi-tier cloud computing systems. "In Cloud Computing (CLOUD), 2011 IEEE International Conference on (pp. 324-331).

17. Ergu, D., Kou, G., Peng, Y., Shi, Y. and Shi, Y. (2013), "The analytic hierarchy process: task scheduling and resource allocation in cloud computing environment. "The Journal of Supercomputing, pp.1-14.

18. Espadas J, Molina A, Jiménez G, Molina M, Ramírez R, Concha D. (2013), "A tenant-based resource allocation model for scaling Software-as-a-Service applications over cloud computing infrastructures. "Future Generation Computer Systems. 29 (1):273-86.

۱۹. Taleb, Tarik, and Adlen Ksentini. "Follow me cloud: interworking federated clouds and distributed mobile networks." Network, IEEE 27.5 (2013): 12-19.

20. Gosavi, A. (2004), "A reinforcement learning algorithm based on policy iteration for average reward: Empirical results with yield management and convergence analysis. " Machine Learning, 55(1), pp.5-29.

21. Lee, J.Y., Lee, J.W. and Kim, S.D. (2009),"A quality model for evaluating software-as-a-service in cloud computing. "In Software Engineering Research, Management and Applications, 2009. SERA'09. 7th ACIS International Conference on (pp. 261-266).