Journal of Soft Computing and Information Technology (JSCIT)
Babol Noshirvani University of Technology, Babol, Iran
Journal Homepage: jscit.nit.ac.ir
Volume 11, Number 2, Summer 2022, pp. 64-75
Received: 12/12/2021, Revised: 08/21/2022, Accepted: 09/18/2022

An approach based on fog nodes collaboration and lottery algorithm for deadline-aware task placement and scheduling in fog computing

Mohammad Amin Safizade[1], Samira Noferesti[2*] and Nik-Mohammad Balouchzahi[3]
1- Faculty of Electrical and Computer Engineering, University of Sistan and Baluchestan, Zahedan, Iran.
2*- Faculty of Electrical and Computer Engineering, University of Sistan and Baluchestan, Zahedan, Iran.
3- Faculty of Electrical and Computer Engineering, University of Sistan and Baluchestan, Zahedan, Iran.
[1]amin.safizade@pgs.usb.ac.ir, [2*]snoferesti@ece.usb.ac.ir, and [3]balouchzahi@ece.usb.ac.ir

Corresponding author's address: Samira Noferesti, Faculty of Electrical and Computer Engineering, University of Sistan and Baluchestan, Zahedan, Iran.

Abstract- Fog computing is a distributed computing paradigm that extends the cloud services to the edge of the network to support real-time and delay-sensitive applications. One of the main issues in fog computing is how to effectively and fairly allocate the restricted resources of fog nodes to users' requests. The limited number of resources, the different requirements of user requests, and latency requirement of delay-sensitive applications have made resource allocation and scheduling challenging. This paper proposes an efficient approach for the placement and scheduling of deadline-aware tasks in fog computing. In the proposed approach, task placement is done with the collaboration of fog nodes and based on the estimation of the completion time of a request in different fog nodes. The lottery algorithm is also used for task scheduling, and the requests are prioritized based on their deadlines. The experimental results show that the combination of fog nodes collaboration for task placement and the lottery algorithm for scheduling reduces the response time and increases the acceptance ratio of user requests. According to the simulation results, the acceptance ratio and the response time of the proposed approach improved by 12.72% and 37.97 *ms*, respectively, compared to the baseline method that uses the FCFS algorithm for task scheduling. Also, in comparison with the existing approaches which use a central controller for fog nodes collaboration, the proposed approach increased the acceptance ratio by 2.57%, and decreased the response time by 20.42 *ms*.

Keywords- Fog computing, Task placement, Deadline-aware task scheduling, Fog nodes collaboration, Lottery algorithm, Fog colony.

مجله علمی پژوهشی رایانش نرم و فناوری اطلاعات
دانشگاه صنعتی نوشیروانی بابل
صفحه مجله: jscit.nit.ac.ir
جلد ۱۱، شماره ۲، تابستان ۱۴۰۱ ، صفحه ۶۴-۷۵
دریافت: ۱۴۰۰/۰۹/۲۱، بازنگری: ۱۴۰۱/۰۵/۳۰، پذیرش: ۱۴۰۱/۰۶/۲۷

ارائه روشی مبتنی بر همکاری گره‌های مه و الگوریتم بخت‌آزمایی برای جایابی و زمانبندی وظایف مهلت‌آگاه در رایانش مه

محمد امین صافی‌زاده[۱]، سمیرا نوفرستی*[۲]، نیک محمد بلوچزهی[۳]
۱- دانشکده مهندسی برق و کامپیوتر، دانشگاه سیستان و بلوچستان، زاهدان، ایران.
*۲- دانشکده مهندسی برق و کامپیوتر، دانشگاه سیستان و بلوچستان، زاهدان، ایران.
۳- دانشکده مهندسی برق و کامپیوتر، دانشگاه سیستان و بلوچستان، زاهدان، ایران.
[1]amin.safizade@pgs.usb.ac.ir, [2*]snoferesti@ece.usb.ac.ir, [3]balouchzahi@ece.usb.ac.ir

* نشانی نویسنده مسئول: سمیرا نوفرستی، خیابان دانشگاه، دانشگاه سیستان و بلوچستان، دانشکده مهندسی برق و کامپیوتر.

چکیده- رایانش‌مه یک الگوی محاسباتی توزیع‌شده است که با گسترش سرویس‌های ابر به لبه شبکه امکان پاسخگویی به کاربردهای حساس به تاخیر و بلادرنگ را فراهم می‌سازد. یکی از مسائل مطرح در رایانش مه چگونگی تخصیص منابع محدود گره‌های مه به درخواست‌های کاربران به شیوه‌ای کارا و عادلانه است. وجود منابع محدود، نیازمندی‌های متفاوت درخواست‌های کاربران و لزوم تخصیص به موقع منابع در کاربردهای حساس به تاخیر، تخصیص و زمانبندی منابع را به مسأله‌ای چالش برانگیز تبدیل کرده است. در این مقاله روشی کارا برای جایابی و زمانبندی وظایف مهلت‌آگاه در رایانش مه پیشنهاد می‌شود. در روش پیشنهادی، جایابی وظایف با همکاری گره‌های مه و بر اساس تخمین زمان اتمام کار یک درخواست در گره‌های مه مختلف انجام می‌شود. زمانبندی وظایف مهلت‌آگاه نیز با استفاده از الگوریتم بخت‌آزمایی و بر مبنای اولویت‌دهی به وظایف بر اساس مهلت زمانی آن‌ها انجام می‌شود. نتایج آزمایشات انجام گرفته نشان می‌دهد که ترکیب همکاری گره‌های مه برای جایابی وظایف و الگوریتم بخت‌آزمایی برای زمانبندی وظایف، باعث کاهش زمان‌پاسخ و افزایش نرخ پذیرش درخواست‌های کاربر می‌شود. بر طبق نتایج حاصل از شبیه‌سازی‌های انجام گرفته، نرخ پذیرش و زمان پاسخ روش پیشنهادی در مقایسه با روش پایه که از الگوریتم خدمت به ترتیب ورود برای زمانبندی درخواست‌ها استفاده می‌کند به ترتیب ۱۲.۷۲ درصد و ۳۷.۹۷ میلی‌ثانیه بهبود داشته است. همچنین روش پیشنهادی در مقایسه با روش‌های موجود که از کنترل‌کننده مرکزی برای هماهنگی گره‌های مه بهره می‌برند، نرخ پذیرش درخواست‌ها را ۲.۵۷ درصد افزایش و زمان پاسخ را ۲۰.۴۲ میلی‌ثانیه کاهش داده است.

واژه‌های کلیدی: رایانش مه، جایابی وظایف، زمانبندی وظایف مهلت‌آگاه، همکاری گره‌های مه، الگوریتم بخت‌آزمایی، کلونی مه

۱- مقدمه

ظهور اینترنت اشیا و فراهم شدن امکان ارتباطات و تعاملات گسترده بین دستگاه‌های هوشمند منجر به تولید حجم زیادی داده غیرهمگن شده است که به عنوان انفجار داده شناخته می‌شود. این حجم عظیم داده‌ها باعث شده است که سرویس‌های ذخیره‌سازی و پردازشی سنتی قادر به پاسخ‌گویی سریع به نیازها و درخواست‌های کاربران نباشند. رایانش ابر به دلیل فراهم کردن فضای ذخیره‌سازی و قدرت پردازشی بالا، یک راهکار موثر برای ذخیره‌سازی و پردازش این حجم از داده‌ها را فراهم می‌کند. با این وجود، رایانش ابری در کاربردهای حساس به تاخیر مانند حمل و نقل هوشمند به دلیل تاخیر ناشی از انتقال اطلاعات به ابر عملکرد موفقی ندارد. برای حل مشکل مذکور رایانش مه[۱] معرفی شده است.

رایانش مه یک معماری محاسباتی توزیع شده است که برای ارائه منعطف سرویس‌های محاسباتی، ذخیره‌سازی و ارتباطی دستگاه-های ناهمگون لبه شبکه را به یکدیگر متصل کرده است. رایانش مه به دلیل نزدیکی به کاربران می‌تواند پاسخگوی کاربردهای حساس به تاخیر و بلادرنگ باشد. برخلاف ابر، دیگر احتیاجی به ارسال همه داده‌ها از هسته شبکه به ابر نیست و این داده‌ها می-توانند در گره‌های مه ذخیره‌سازی و پردازش شوند. در نتیجه پرداختن به درخواست‌های کاربران می‌تواند به راحتی و با تاخیر بسیار کمتر نسبت به ابر فراهم گردد.

با توجه به محدود بودن انرژی و منابع پردازشی گره‌های مه، یکی از نیازهای رایانش مه جایابی و زمانبندی وظایف به شیوه‌ای عادلانه، کارا و با توجه به اولویت درخواست‌های کاربران است. اگرچه مسأله زمانبندی وظایف در شبکه‌های سنتی از دیرباز مطرح بوده و الگوریتم‌های متعددی برای آن مطرح شده است، ویژگی‌ها و نیازمندی‌های خاص رایانش مه نظیر حجم عظیم درخواست‌ها و نیاز به زمان پاسخ معقول به ویژه برای درخواست‌های حساس به تاخیر باعث شده است استفاده از الگوریتم‌های زمانبندی سنتی در رایانش مه ناکارامد و گاهاً ناممکن باشد.

در مقایسه با رایانش ابری، تحقیقات اندکی در زمینه زمانبندی وظایف در رایانش مه انجام گرفته است که اغلب آنها به نیازمندی-های متفاوت درخواست‌ها و همکاری گره‌های مه برای پاسخ‌گویی به درخواست‌های کاربران توجه نداشته‌اند. همچنین اکثر مطالعات موجود بر روی بهبود یک معیار کارایی تمرکز داشته و کارایی سیستم را با معیارهای متضاد ارزیابی نکرده‌اند. هدف این مقاله ارائه روشی کارا از نظر زمان پاسخ و نرخ پذیرش درخواست‌ها برای جایابی و زمانبندی وظایف در رایانش مه است. به طور خلاصه نوآوری‌های روش پیشنهادی عبارتند از:

- ارائه روشی جدید برای جایابی درخواست‌های مهلت‌آگاه در رایانش مه با در نظر گرفتن معماری سلسله‌مراتبی رایانش مه، نقش کمکی ابر و وضعیت ازدحام درخواست‌ها در گره‌های مه و با هدف کاهش زمان پاسخ و افزایش نرخ پذیرش درخواست‌های کاربران: در روش پیشنهادی با توجه به مهلت زمانی یک درخواست و وضعیت ازدحام درخواست‌ها در گره مه دریافت کننده درخواست، درباره اجرای درخواست در گره مه یا ارسال آن به یکی از گره-های مه همسایه یا ابر تصمیم گرفته می‌شود.
- تعریف یک همسایگی از گره‌های مه به نام کلونی مه که به منظور جایابی درخواست‌های کاربران بر اساس تخمین زمان اتمام کار درخواست‌ها با یکدیگر همکاری می‌کنند. در روش پیشنهادی تصمیم‌گیری در رابطه با انتخاب بهترین گره برای اجرای درخواست کاربر (ابر یا یکی از گره‌های مه در کلونی)، بر اساس ویژگی‌های درخواست (اولویت و مهلت زمانی آن) و ویژگی‌های گره‌های مه (توان پردازشی، تعداد درخواست‌ها و ...) انجام می‌شود که می‌تواند به توزیع بار بین گره‌های مه و جلوگیری از ازدحام درخواست‌ها در یک گره مه و به تبع آن عدم نقض مهلت زمانی درخواست‌های منتظر در صف گره مه کمک کند. در روش پیشنهادی بر خلاف کارهای پیشین از کنترل‌کننده مرکزی برای هماهنگی گره‌های مه استفاده نشده است و گره‌های مه با همکاری یکدیگر و از طریق تبادل پیام درباره مکان اجرای درخواست‌ها تصمیم‌گیری می‌کنند که مشکل گلوگاه شدن کنترل-کننده مرکزی در روش‌های پیشین را برطرف می‌کند.
- بکارگیری الگوریتم بخت‌آزمایی برای زمانبندی مبتنی بر اولویت وظایف: در زمانبندی مبتنی بر اولویت امکان در نظر گرفتن دسته‌های متفاوت برنامه‌های کاربردی با نیاز-های متفاوت فراهم می‌شود. در این مقاله به طور خاص اولویت بر اساس مهلت زمانی درخواست‌ها تعیین می‌شود که منجر به افزایش نرخ پذیرش و کاهش زمان پاسخ درخواست‌ها به ویژه برای درخواست‌های با اولویت بالا (درخواست‌های دارای مهلت زمانی محدود و حساس به تاخیر) می‌شود.

نتایج آزمایشات انجام گرفته نشان می‌دهد که روش پیشنهادی برای زمانبندی وظایف در مقایسه با روش پایه خدمت به ترتیب ورود (FCFS) و نیز روش مبتنی بر کنترل‌کننده مرکزی برای همکاری گره‌های مه [۱، ۲]، کارایی بالاتری بر اساس معیارهای نرخ پذیرش درخواست‌ها و زمان پاسخ دارد.

ادامه مقاله به‌صورت زیر سازماندهی شده است. در بخش ۲ به طور خلاصه تحقیقات پیشین در زمینه جایابی و زمانبندی وظایف در رایانش مه معرفی می‌شود. در بخش ۳ جزئیات روش پیشنهادی برای جایابی و زمانبندی وظایف مهلت‌آگاه تشریح می‌گردد. در فصل ۴ نتایج ارزیابی کارایی روش پیشنهادی ارائه می‌شود. در پایان بخش ۵ نتیجه‌گیری می‌باشد.

۲- مرور تحقیقات پیشین

در حال حاضر تحقیقات متعددی در رابطه با مدیریت منابع و زمانبندی وظایف در رایانش ابر وجود دارد [۵-۳]، اما مسأله جایابی و زمانبندی وظایف در رایانش مه در مراحل ابتدایی خود

است. در برخی از تحقیقات پیشین از الگوریتم‌های فرا ابتکاری مانند الگوریتم آتش‌بازی [۶]، کلونی مورچه‌ها [۷]، پرواز پرندگان [۸]، الگوریتم نهنگ [۹] و بهینه‌سازی مبتنی بر اکوسیستم [۱۰] برای زمانبندی وظایف در رایانش مه استفاده شده است. روش‌های فرا ابتکاری برای تخصیص منابع گره‌های مه به درخواست‌های کاربران در تکرارهای ابتدایی به صورت تصادفی عمل کرده و به تدریج سعی در بهبود راه‌حل با توجه به یک تابع شایستگی از پیش تعریف شده دارند. روش‌های فرا ابتکاری بهینگی راه‌حل پیشنهادی را تضمین نمی‌کنند. همچنین در اغلب پژوهش‌های این دسته به نیازها و اولویت‌های مختلف درخواست‌های کاربران توجه نشده است.

در برخی دیگر از کارهای موجود، زمانبندی مبتنی بر اولویت مورد استفاده قرار گرفته است. در این روش‌ها برای برخی از درخواست-ها نظیر درخواست‌های حساس به تاخیر اولویت بیشتری در نظر گرفته می‌شود [۱۱، ۱۲]. برای نمونه در [۱۱] درخواست‌ها بر اساس اولویت به دو دسته درخواست‌های واقعی (حساس به تاخیر) و نرم (غیر حساس به تاخیر) دسته‌بندی شده‌اند. در این روش، زمانبندی توسط سرور اصلی مه با استفاده از الگوریتم RTES[۲] با اهداف توازن بار در محیط مه، اجرای درخواست‌ها در مهلت تعیین شده، افزایش گذردهی، بهینه‌سازی شبکه و نگهداری موقت داده-های مورد نیاز کاربر، صورت می‌گیرد. مشکل اصلی روش‌های مبتنی بر اولویت گرسنگی است که ممکن است درخواست‌های با اولویت کم برای مدت نامشخص منتظر بمانند.

تحقیقاتی نیز در زمینه همکاری گره‌های مه برای حل مسائل مختلف در رایانش مه انجام گرفته است. برای نمونه در [۱۳] یک استراتژی همکاری برای ارسال بارکاری پیشنهاد شده است. در این طرح، یک گره مه به جای متکی بودن به مراکز داده ابر برای پردازش درخواست‌ها، می‌تواند تمام حجم کار پردازشی خود یا بخشی از آن را به گره‌های مه همسایه ارسال کند. با این وجود، تحقیقات انجام گرفته در زمینه جایابی و زمانبندی وظایف مبتنی بر همکاری و ارتباطات بین گره‌های مه اندک است. در [۱، ۲] از الگوریتم ژنتیک برای جایابی سرویس در کاربردهای اینترنت اشیاء استفاده شده است. در این روش گره‌های مه در تعدادی کلونی قرار می‌گیرند. هر کلونی مه برای اجرای سرویس و مدیریت منابع خود، دارای یک گره کنترل‌کننده مرکزی است. ارتباط یک کلونی با دیگر کلونی‌ها و همچنین ارتباط کلونی با ابر از طریق این گره صورت می‌پذیرد.

در [۱۴] یک استراتژی جایابی سرویس در رایانش مه معرفی شده است که سعی در قرار دادن سرویس‌های حساس به تاخیر در نزدیک کاربر دارد. در این مقاله دستگاه‌های مه به تعدادی خوشه تقسیم می‌شوند که هر خوشه یک کلونی نامیده می‌شود و دارای یک گره کنترل کننده است. به منظور کاهش هزینه و زمان پاسخ درخواست‌های کاربر، یک الگوریتم هیوریستیک در گره کنترل کننده کلونی اجرا می‌شود که درباره محل قرارگیری سرویس‌ها بر روی گره‌های محاسباتی تصمیم می‌گیرد. ایده اصلی این الگوریتم برای کاهش تاخیر، مرتب‌سازی برنامه‌های کاربردی به ترتیب صعودی مهلت زمانی و مرتب‌سازی گره‌های محاسباتی به ترتیب نزدیکی به کاربران در کلونی مربوطه و قرار دادن سرویس‌های حساس به تاخیر بر روی منابع نزدیک به کاربر است. بدین ترتیب سرویس‌های برنامه‌های کاربردی حساس به تاخیر تا جایی که ممکن است بر روی گره کنترلی اجرا می‌شوند و اگر این گره منابع کافی برای اجرای سرویس را نداشت، سرویس به گره محاسباتی بعدی در لیست ارسال می‌شود.

در [۱۵] تاثیر ساختارهای مختلف برای کلونی مه بر روی کارایی روش پیشنهاد شده در [۱] بررسی شده است. در این مقاله هر کلونی مه دارای یک کنترل کننده مرکزی است که وظیفه توزیع سرویس‌ها بر روی گره‌های مه را دارد. یک درخواست متشکل از تعدادی سرویس همکاری کننده در نظر گرفته می‌شود که هر کدام می‌توانند بر روی یک گره مه متمایز در کلونی اجرا شوند. همچنین بر اساس شاخص‌های مختلف به تقسیم گره‌های مه در یک کلونی و تعیین کنترل‌کننده مرکزی هر کلونی پرداخته شده است.

در اغلب تحقیقات موجود برای همکاری گره‌های مه از یک کنترل‌کننده مرکزی استفاده شده است [۱،۲، ۱۷-۱۴]. در این روش‌ها، کلیه درخواست‌های کاربران در ابتدا به گره کنترل‌کننده مرکزی ارسال می‌شوند و پس از تجزیه و تحلیل، جایابی می‌گردند. این امر باعث گلوگاه شدن کنترل‌کننده مرکزی در هنگام ازدحام و به تبع آن افزایش زمان پاسخ و کاهش کیفیت سرویس می‌شود. در این مقاله یک روش مبتنی بر اولویت برای زمانبندی معرفی می‌شود که برای جایابی درخواست‌ها از همکاری گره‌های یک کلونی استفاده می‌کند. در روش پیشنهادی برای کلونی مه گره کنترل‌کننده مرکزی در نظر گرفته نشده است و هر گره کلونی خود در رابطه با جایابی درخواست‌هایش تصمیم‌گیری می‌کند. در جدول ۱ خلاصه تحقیقات پیشین گزارش شده است.

جدول ۱: خلاصه تحقیقات پیشین در زمینه جایابی و زمانبندی وظایف در رایانش مه.

مرجع	هدف	دسته	همکاری گره‌های مه	کنترل‌کننده مرکزی	ویژگی‌ها
[۶-۱۰]	زمانبندی وظایف/ تخصیص منابع با هدف کاهش زمان اتمام کار و هزینه	فرا ابتکاری	ندارد	-	عملکرد تصادفی جهت فرار از بهینه محلی، توجه به زمان اجرا و مهلت زمانی وظایف در زمانبندی عدم تضمین یافتن پاسخ بهینه، مناسب نبودن الگوریتم برای شبکه‌های بزرگ با تعداد گره‌های بالا، عدم توجه به فاصله‌ی میان گره‌های مه و نیازمندی‌های درخواست‌ها، سرعت همگرایی کم
[۱۱]	زمانبندی وظایف با هدف توازن بار	مبتنی بر اولویت	ندارد	-	پیاده‌سازی ساده، اولویت دادن به درخواست‌ها بر اساس مهلت زمانی آنها تمرکز بر توازن بار و عدم در نظر گرفتن دیگر معیارهای کارایی مانند زمان پاسخ و نرخ پذیرش
[۱۲]	زمانبندی وظایف با هدف کاهش زمان پاسخ و هزینه	مبتنی بر اولویت	دارد	ندارد	اولویت دادن به درخواست‌ها بر اساس مهلت زمانی آنها درنظر نگرفتن احتمال گرسنگی در صف‌های اولویت پایین
[۱۴]	جایابی سرویس با هدف کاهش زمان پاسخ و هزینه	مبتنی بر اولویت	دارد	دارد	اولویت دادن به درخواست‌ها بر اساس مهلت زمانی آنها و اولویت دهی به منابع بر حسب فاصله آنها تا کاربر انعطاف پذیری کم، عدم توجه به تاخیرهای ارتباطی و فاصله میان گره‌های مه، گلوگاه شدن گره کنترل کننده
[۱۵]	تعیین ساختار کلونی برای جایابی کاراتر سرویس	فرا ابتکاری	دارد	دارد	توجه به شاخص‌های مختلف برای خوشه‌بندی گره‌های مه، در نظر گرفتن فاصله گره‌های مه گلوگاه شدن گره‌های کنترل کننده
[۱۶]	جایابی وظایف با هدف کاهش زمان اتمام کار و تاخیر	مبتنی بر اولویت	دارد	دارد	معماری چهار لایه برای رایانش مه، اولویت دادن به درخواست‌ها بر اساس مهلت زمانی آنها عدم توجه به تاخیرهای ارتباطی و فاصله میان گره‌های مه، گلوگاه شدن گره‌های کنترل کننده
[۱۷]	زمانبندی وظایف/ تخصیص منابع	فرا ابتکاری	دارد	دارد	خوشه‌بندی وظایف بر اساس فضای ذخیره‌سازی، زمان اجرا و پهنای باند موردنیاز سرعت همگرایی کم، عدم تضمین یافتن پاسخ بهینه، گلوگاه شدن گره‌های کنترل کننده
[۱، ۲]	جایابی و زمانبندی وظایف با هدف کاهش زمان پاسخ و ارضا محدودیت منابع	فرا ابتکاری	دارد	دارد	توزیع سرویس‌های یک درخواست بر روی گره‌های مه یک کلونی، در نظر گرفتن تاخیر ارتباطی بین گره‌ها گلوگاه شدن گره‌های کنترل کننده
روش پیشنهادی	جایابی و زمانبندی وظایف با هدف کاهش زمان پاسخ و افزایش نرخ پذیرش	مبتنی بر اولویت	دارد	ندارد	اولویت دهی به درخواست‌ها با توجه به مهلت زمانی آن‌ها، جلوگیری از ازدحام درخواست‌ها در یک گره مه، توجه به فاصله گره‌های مه و تاخیر ارتباطی بین آنها، رفع مشکل گلوگاه شدن گره‌های کنترل کننده

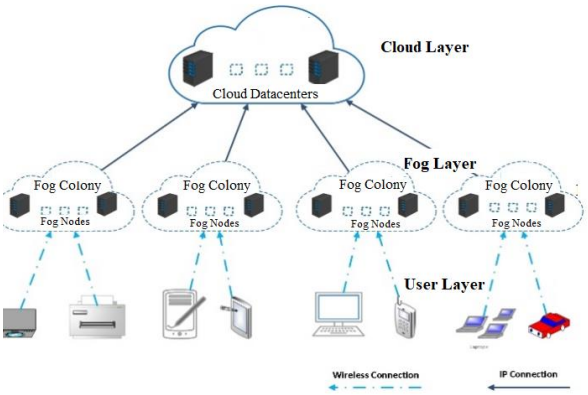
۳- روش پیشنهادی

در این بخش در ابتدا معماری مورد استفاده توصیف می گردد. سپس مدل سیستم پیشنهادی ارائه می‌گردد و در پایان به شرح روش پیشنهادی برای زمانبندی وظایف مهلت‌آگاه پرداخته می‌شود.

۳-۱- معماری رایانش مه

معماری مورد استفاده در این تحقیق، معماری سه لایه رایانش مه [۱۸] می‌باشد که در شکل ۱ نشان داده شده است. در معماری سه لایه، لایه ابر شامل مراکز داده ابر در مقیاس بزرگ است که این واحدهای محاسباتی با کارایی بالا معمولاً با فاصله زیاد از کاربران واقع شده‌اند.

لایه دوم، لایه مه نامیده می‌شود و توسط شبکه پروتکل اینترنت[۳] به لایه ابر متصل است. لایه مه شامل مجموعه‌ای از گره‌های مه است که در مکان‌هایی نزدیک به کاربران به طور گسترده مستقر هستند.



شکل ۱: معماری سلسله‌مراتبی رایانش مه.

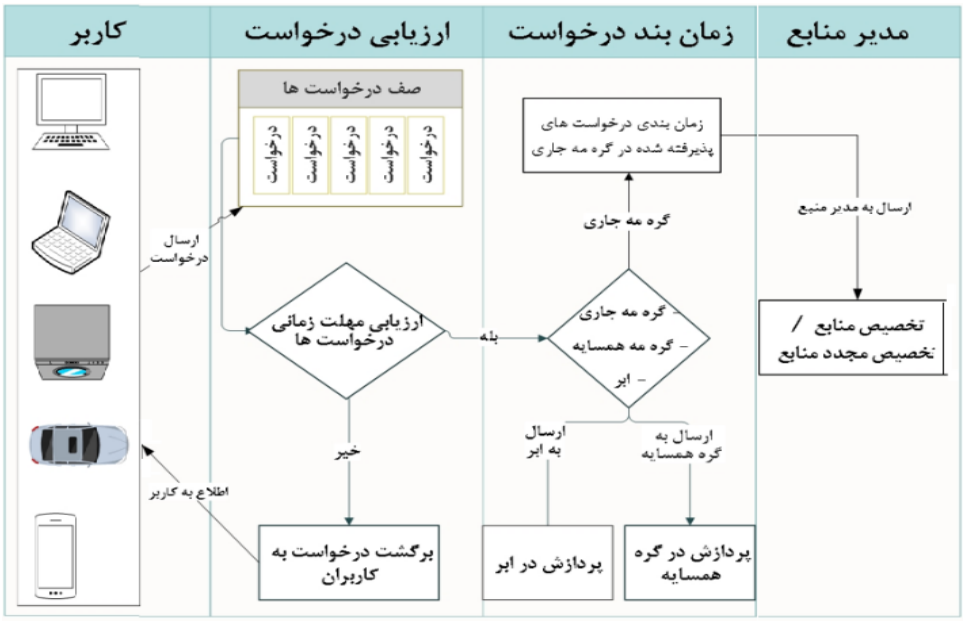
هر گره مه از منابع محاسباتی و انرژی محدودی برخوردار است. بدین دلیل جایابی مناسب درخواست‌های کاربران تاثیر چشمگیری بر بهره‌وری منابع گره‌های مه دارد. در این مقاله لایه دوم به صورت کلونی مه در نظر گرفته شده است. هر گره مه در کلونی، می‌تواند با دیگر گره‌های کلونی ارتباط برقرار کند.

آخرین لایه، لایه کاربر است که به وسیله تکنولوژی بی‌سیم با لایه مه در ارتباط است. لایه کاربر درخواست‌های خود را برای اجرا به لایه مه ارسال می‌کند. در ابتدا درخواست ارسال شده توسط یک کاربر به نزدیک‌ترین گره مهی که کاربر در منطقه تحت پوشش آن قرار دارد، داده می‌شود. پس از آن با همکاری گره‌های مه یک کلونی، درخواست جایابی شده و درباره محل اجرای آن تصمیم-گیری می‌شود.

۳-۲- مدل سیستم

مدل سیستم پیشنهادی برای تخصیص منابع (به طور خاص پردازنده) در رایانش مه در شکل ۲ نشان‌داده شده است. این سیستم شامل N کاربر با توان پردازشی پایین و مجموعه‌ای از M گره مه برای ارائه سرویس به کاربرانی که در نزدیکی آنها قرار دارند، است. همچنین ابر در فاصله‌ای دور از کاربران برای ارائه سرویس‌های مختلف رایانش‌ابر قرار دارد و منابع محاسباتی، ذخیره‌سازی و شبکه را برای گره‌های مه فراهم می‌کند.

مطابق شکل ۲، هر گره‌مه دارای سه مؤلفه می‌باشد که عبارتند از ارزیابی درخواست، زمانبند درخواست و مدیر منابع. مولفه ارزیابی درخواست وظیفه تصمیم‌گیری در رابطه با پذیرش یا عدم پذیرش درخواست‌ها با توجه به مهلت زمانی آنها را دارد. در این مقاله یک سیستم بلادرنگ در نظر گرفته شده است، به این معنی که وظایف بایستی قبل از به اتمام رسیدن مهلت زمانی آنها اجرا شوند. مؤلفه ارزیابی درخواست، درخواست‌ها را به ترتیب ورود بررسی می‌کند؛ اگر گره مه دریافت کننده درخواست، یکی از همسایه‌های آن و یا ابر قادر به پاسخ‌گویی به درخواست فعلی با توجه به مهلت زمانی آن نباشند، پیامی به کاربر داده می‌شود که درخواستی دیگری با مهلت زمانی جدید ارسال کند. اگر درخواست پذیرفته شود، مؤلفه ارزیابی درخواست، درخواست را به مؤلفه زمانبند درخواست می‌دهد.



شکل ۲: مدل سیستم پیشنهادی برای تخصیص منابع.

مولفه زمانبند درخواست، وظیفه جایابی و زمانبندی درخواست‌ها را دارد. این مؤلفه مشخص می‌کند که درخواست بایستی بر روی گره‌مه جاری (گره دریافت کننده درخواست)، یکی از گره‌های همسایه و یا ابر اجرا شود. سپس درخواست‌هایی که قرار است در گره مه جاری اجرا شوند را زمانبندی کرده و به مؤلفه مدیر منابع تحویل می‌دهد.

مولفه مدیر منابع، منابع مورد نیاز برای اجرای درخواست را به آن اختصاص می‌دهد. درخواست‌ها در این مقاله به عنوان کوچکترین واحد پردازش درنظر گرفته شده‌اند، بنابراین قابل تقسیم به چند زیر وظیفه برای اجرا بر روی گره‌های مختلف نمی‌باشند.

۳-۳- روش پیشنهادی برای جایابی و زمانبندی وظایف

الگوریتم زمانبندی پیشنهادی یک الگوریتم مبتنی بر اولویت ایستا است که در ابتدای بازه‌های زمانی اجرا می‌شود و تنها درخواست-هایی که تا آن لحظه به سیستم رسیده‌اند را در نظر می‌گیرد. اولویت درخواست‌ها با توجه به مهلت زمانی آنها تعیین می‌شود. به این صورت که درخواست‌های کاربر بسته به میزان حساس به تاخیر بودن آنها به ۷ کلاس تقسیم می‌شوند. هر چه درخواست حساسیت بیشتری به تاخیر داشته باشد اولویت بالاتری دارد. همچنین درخواست‌های ورودی دارای مهلت زمانی هستند و اگر نتوان درخواست را در مهلت مقرر انجام داد، درخواست رد می‌شود.

در ابتدای هر بازه زمانی، مولفه زمانبند درخواست در هر گره درباره جایابی درخواست‌های دریافت شده در آن گره تصمیم‌گیری می‌کند. درخواست می‌تواند در گره مه جاری اجرا شود یا به یکی از گره‌های همسایه یا به ابر ارسال شود. در روش پیشنهادی، جایابی درخواست‌ها مبتنی بر همکاری گره‌های مه است. برای این منظور گره‌های مه به صورت کلونی در نظر گرفته شده‌اند. گره‌های یک کلونی که همسایه نامیده می‌شوند می‌توانند درخواست‌های خود را مبادله کنند. هر کلونی شامل پنج گره مه و فاقد کنترل-کننده مرکزی است (در بخش ۴-۲ روش تعیین تعداد بهینه گره-های مه برای هر کلونی تشریح شده است).

نحوه انتخاب محل مناسب برای اجرای درخواست فعلی به این صورت است که زمان اتمام کار درخواست در صورت اجرا در گره مه جاری، هر یک از گره‌های دیگر کلونی و ابر بر اساس فرمول (۱) تخمین زده می‌شود و درخواست برای اجرا به گرهی که کمترین زمان اتمام کار را داشته باشد، ارسال می‌شود.

$$TT_{(r,i,t)} = ND_{(i,j,t)} + RT_{(r,i,t)} + WT_{(r,i,t)} \qquad (۱)$$

که در آن $TT_{(r,i,t)}$ نشان‌دهنده زمان اتمام کار تخمین زده شده برای اجرای درخواست r در گره مه i در بازه زمانی t می‌باشد. $ND_{(i,j,t)}$ تاخیر انتها به انتهای ارسال پیام به گره مه i از گره همسایه j در بازه زمانی t و $RT_{(r,i,t)}$ زمان‌اجرای درخواست r را در گره i در بازه زمانی t نشان‌می‌دهد. همچنین $WT_{(r,i,t)}$ نشان‌دهنده زمان‌انتظار تخمین زده شده برای درخواست r در صورت اجرا در گره مه i می‌باشد. برای محاسبه زمان اجرای درخواست از فرمول (۲) استفاده می‌شود.

$$RT_{(r,i,t)} = \frac{L_r}{MIPS_{(i,t)}} \qquad (۲)$$

که L_r طول درخواست بر حسب تعداد دستورالعمل و $MIPS_{(i,t)}$ توان‌پردازشی گره i در بازه زمانی t را نشان‌می‌دهند. برای تخمین زمان‌انتظار درخواست‌ها از فرمول (۳) استفاده می‌شود.

$$WT_{(r,i,t)} = AWT_{(r,i,t-1)} * Q \qquad (۳)$$

که $AWT_{(r,i,t-1)}$ نشان‌دهنده میانگین زمان‌انتظار درخواست‌های با اولویت برابر با درخواست فعلی در بازه $(t-1)$ در گره i می-باشد. همچنین Q ضریب ازدحام را مشخص می‌کند. به دلیل حائز اهمیت بودن اولویت درخواست‌ها در روش پیشنهادی و به منظور تخمین زمان انتظار متناسب با اولویت درخواست ارسالی، برای محاسبه میانگین زمان انتظار گره، تنها درخواست‌هایی که اولویت مساوی با درخواست جاری کاربر دارند، درنظرگرفته شده است. $AWT_{(r,i,t-1)}$ به تنهایی معیار مناسبی برای انتخاب بهترین گره همسایه نیست. زیرا امکان این وجود دارد که گره در بازه زمانی $(t-1)$ دچار ازدحام شده و میانگین زمان‌انتظار بالایی داشته باشد. بدین دلیل در بازه t درخواستی برای گره ارسال نمی‌شود و حتی در صورت ارسال درخواست، تعداد آنها محدود می‌باشد. از سوی دیگر، این امکان وجود دارد که یک گره در بازه زمانی $(t-1)$ تعداد محدودی درخواست داشته باشد اما در بازه زمانی t دچار ازدحام شود. در نتیجه با ارسال درخواست‌های گره‌های همسایه به آن، ازدحام آن بیشتر می‌شود که خود منجر به افزایش زمان‌پاسخ و کاهش نرخ‌پذیرش درخواست‌ها می‌گردد. بنابراین برای مدنظر قراردادن تاثیر ازدحام شبکه بر روی سیستم، از متغیر ضریب ازدحام استفاده شده است که مطابق فرمول (۴) محاسبه می‌شود.

$$Q = \frac{RR_{(i,t)}}{RR_{(i,t-1)}} \qquad (۴)$$

که $RR_{(i,t)}$ نسبت درخواست‌های با اولویت بیشتر یا مساوی درخواست جاری به کل درخواست‌های گره i در زمان t و $RR_{(i,t-1)}$ نسبت درخواست‌های با اولویت بیشتر یا مساوی

درخواست جاری به کل درخواست‌های گره i در زمان $(t-1)$ را نشان‌می‌دهد. برای محاسبه $RR_{(i,t)}$ از فرمول (۵) استفاده می‌شود.

$$RR_{(i,t)} = \frac{NR_{(r,i,t)}}{N_{(i,t)}} \quad (۵)$$

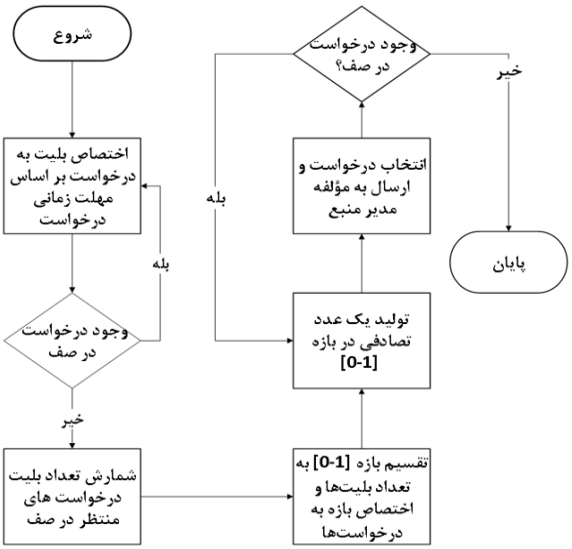
که $NR_{(r,i,t)}$ نشان‌دهنده تعداد درخواست‌های با اولویت بیشتر یا مساوی درخواست r در بازه زمانی t در گره i می‌باشد. همچنین $N_{(i,t)}$ تعداد کل درخواست‌ها در بازه زمانی t در گره i را مشخص می‌کند.

برای محاسبه فرمول‌های فوق، در ابتدای هر بازه زمانی، هر گره در کلونی پیامی حاوی توان پردازشی فعلی و میانگین زمان‌انتظار در بازه قبلی خود را برای دیگر همسایه‌ها ارسال می‌کند. از آنجا که در ابتدای کار سیستم، گره‌های مه، میانگین زمان انتظار خود را ندارند، این میانگین برای همه گره‌ها ثابت درنظر گرفته شده است. بدین ترتیب با استفاده از فرمول‌های توصیف شده می‌توان زمان اتمام کار یک درخواست را برای هر یک از گره‌های کلونی تخمین زد. از آنجا که فرمول‌ها شامل عملیات ساده ریاضی هستند و توان پردازشی بسیار پایینی نیاز دارند، نگرانی برای سربار پردازشی این محاسبات وجود ندارد.

در این مقاله ابر با منابع بینهایت درنظر گرفته شده است که همیشه قادر به پاسخ‌گویی به درخواست‌ها می‌باشد. بنابراین گره مه جاری با داشتن تاخیر انتقال اطلاعات به ابر قادر است زمان اتمام کار درخواست در صورت ارسال به ابر را نیز تخمین بزند. در پایان گره دریافت کننده درخواست با مقایسه زمان اتمام کار تخمین زده شده برای خود، گره‌های مه همسایه و ابر، درباره مکان پردازش درخواست تصمیم گرفته و درخواست را برای گرهی که کمترین زمان اتمام کار را داشته باشد ارسال می‌کند.

پس از جایابی درخواست‌ها در ابتدای هر بازه زمانی، زمانبندی درخواست‌های موجود در صف یک گره مه، توسط الگوریتم بخت‌آزمایی صورت می‌پذیرد. ایده اصلی این است که به درخواست‌های کاربران با توجه به مهلت زمانی آنها بلیت بخت-آزمایی داده شود. بدین صورت که مؤلفه زمانبند درخواست، به هر درخواستی که مهلت زمانی آن در بازه [۳۵۰-۲۵۰] میلی‌ثانیه باشد ۷ بلیت اختصاص می‌دهد. به طور مشابه به درخواست‌هایی که مهلت زمانی آنها در بازه‌های [۴۵۰-۳۵۰]، [۵۵۰-۴۵۰]، [۷۵۰-۵۵۰]، [۱۰۰۰-۷۵۰] و [۱۲۵۰-۱۰۰۰] باشد به ترتیب ۶، ۵، ۴، ۳ و ۲ بلیت و به بقیه درخواست‌ها ۱ بلیت تخصیص می-دهد. به بیانی دیگر، هر چه درخواست حساسیت بیشتری به تاخیر داشته باشد بلیت بیشتری را به خود اختصاص می‌دهد.

زمانبند درخواست یک بلیت بخت‌آزمایی را به تصادف انتخاب کرده و درخواستی که آن بلیت را در دست دارد را برای مؤلفه مدیر منابع ارسال می‌کند. این مؤلفه، منابع لازم را به درخواست اختصاص داده و بعد از اتمام درخواست، منابع آن را آزاد می‌کند. روندنمای الگوریتم پیشنهادی برای زمانبندی وظایف در شکل ۳ نشان داده شده است.



شکل ۳: روندنمای زمانبندی وظایف با استفاده از الگوریتم بخت-آزمایی.

۴- نتایج

در این بخش ابتدا جزئیات شبیه‌سازی انجام گرفته تشریح می-گردد و سپس به ارزیابی کارایی روش پیشنهادی برای زمانبندی وظایف پرداخته می‌شود.

۴-۱- جزئیات شبیه‌سازی

در این مقاله، برای شبیه‌سازی معماری سه لایه از شبیه‌ساز کلودسیم[۴] استفاده شده است [۱۹]. برای شبیه‌سازی لایه ابر، از یک مرکز داده متمرکز استفاده شده است. برای شبیه‌سازی لایه دوم معماری، ۴ کلونی مه تعریف شده است که هر کلونی ۵ گره مه و هر گره مه یک مرکز داده دارد. جدول ۲ مشخصات مراکز داده را که بر اساس مقالات [۲۰، ۲۱] تعیین شده است، نشان می‌دهد. برای لایه کاربر در نرم‌افزار کلودسیم نیاز به تعریف موجودیت خاص نداریم.

جدول ۲: مشخصات پیکربندی گره‌های مه و ابر.

گره	نرخ واحد پردازشی (MIPS)	حافظه (MB)	پهنای باند کانال ارتباطی با لایه بالا (Kbps)	پهنای باند کانال ارتباطی با لایه پایین (Kbps)	پهنای باند کانال ارتباطی با گره‌های مه مجاور (Kbps)
مه	۲۸۰۰	۴۰۰۰	۱۰۰۰۰	۱۰۰۰۰	۱۰۰۰۰۰
ابر	۴۴۸۰۰	۴۰۰۰۰	۱۰۰	۱۰۰۰۰	-

جدول ۳ تاخیر بین گره‌های مختلف را نشان می‌دهد.

جدول ۳: تاخیر بین گره‌های مختلف.

گره	تاخیر (میلی‌ثانیه)
کاربر – کلونی	۲
گره‌های کلونی	۲۰ - ۲۵
گره‌های مه- ابر	۱۰۵ - ۱۴۰

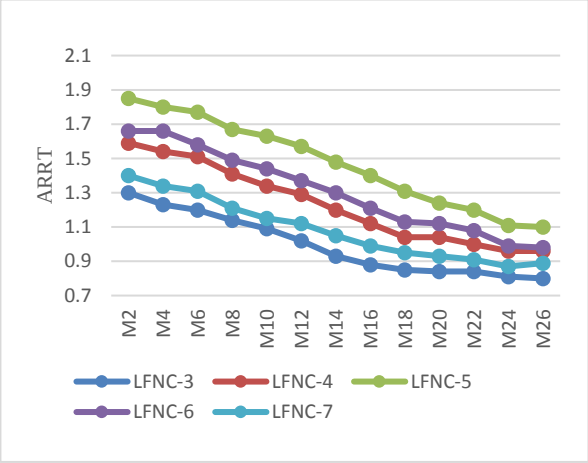
ورودی مدل پیشنهادی، درخواست‌هایی است که توسط کاربران در زمان‌های مختلف وارد می‌شوند. هر درخواست دارای ویژگی‌های شماره درخواست، طول درخواست، شماره کلونی، مهلت زمانی، تعداد بلیت بخت‌آزمایی (با توجه به مهلت زمانی تعیین می‌شود) و زمان ارسال می‌باشد. در شبیه‌سازی انجام گرفته، درخواست‌ها توسط تابع پواسون با نرخ ورودی $\lambda = 0.5$ ایجاد شده‌اند. تعداد درخواست‌های تولید شده ۸۶۱ درخواست است که در ۲۶ بازه اجرا شده‌اند.

در این مقاله، مهلت زمانی نیمی از درخواست‌ها در بازه [۲۵۰-۷۵۰] میلی‌ثانیه در نظر گرفته شده است که درخواست‌های حساس به تاخیر هستند. همچنین به دلیل اینکه روش پیشنهادی مبتنی بر تخمین زمان اتمام کار برای درخواست‌ها است و نتایج حاصل در هر بار شبیه‌سازی تغییر می‌کند، در کلیه آزمایشات این بخش، نتایج ارائه شده میانگین ۱۰ بار اجرای روش پیشنهادی است.

۴-۲- تعیین تعداد گره‌های مه در هر کلونی

یکی از مسائل مطرح در روش پیشنهادی، تعیین تعداد گره‌های مه در هر کلونی است. اگر تعداد گره‌ها کم در نظر گرفته شود، به دلیل محدود شدن منابع کلونی، زمان پاسخ افزایش و نرخ پذیرش درخواست‌ها کاهش می‌یابد. اگر تعداد گره‌های یک کلونی را زیاد در نظر بگیریم، به دلیل افزایش تعاملات بین گره‌ها و افزایش حجم محاسبات برای هر درخواست، کارایی سیستم کاهش پیدا می‌کند. بنابراین در آزمایش اول سعی شده است تعداد گره‌های مه در هر کلونی به گونه‌ای تعیین شود که علاوه بر داشتن حداقل زمان پاسخ، بیشترین نرخ پذیرش درخواست‌ها حاصل شود. به منظور بررسی تاثیر همزمان معیارهای نرخ پذیرش و زمان پاسخ، معیار جدیدی تعریف شده است که از تقسیم نرمالایز شده نرخ پذیرش درخواست‌ها بر نرمالایز شده میانگین زمان پاسخ درخواست‌ها به‌دست می‌آید. این معیار را ARRT[5] نام‌گذاری کرده‌ایم. هرچه معیار ARRT بیشتر باشد، کارایی سیستم نیز بالاتر خواهد بود.

به منظور تعیین تعداد گره‌های مه در هر کلونی، ابتدا یک مجموعه (متفاوت از مجموعه داده مورد استفاده برای ارزیابی کارایی روش پیشنهادی) شامل ۱۲۳۸ درخواست تولید شد. سپس روش ارائه شده در این مقاله که مبتنی همکاری گره‌های مه و استفاده از الگوریتم بخت‌آزمایی برای جایابی و زمانبندی درخواست‌ها است و LFNC[6] نامیده می‌شود را با تعداد مختلف گره‌های مه در هر کلونی پیاده‌سازی کردیم. شکل ۴ نمایانگر کارایی روش پیشنهادی بر اساس معیار ARRT با تعداد گره‌های مه متفاوت در هر کلونی است. M_i نمایانگر بازه زمانی [i-0] است.



شکل ۴: کارایی روش پیشنهادی (LFNC) با تعداد گره‌های مه مختلف در هر کلونی.

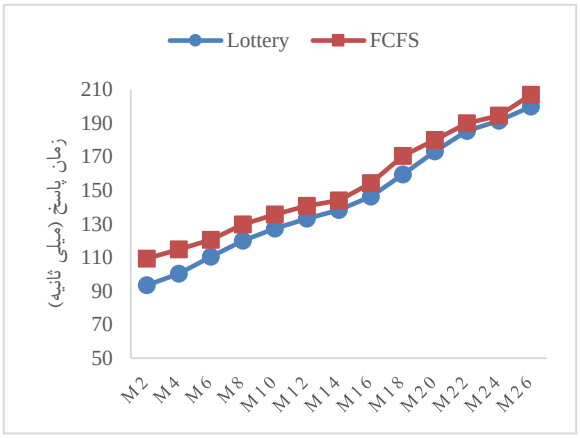
نتایج به دست آمده در شکل ۴ نشان‌دهنده این است که وجود ۵ گره مه در هر کلونی بیشترین کارایی سیستم را به همراه دارد. دلیل اصلی کاهش معیار ARRT برای شبیه‌سازی سیستم با ۶ و ۷ گره مه در کلونی، افزایش تعداد پیام‌های ردوبدل شده برای همکاری گره‌های مه و افزایش محاسبات انجام شده به منظور یافتن بهترین گره همسایه است. افزایش پیام‌ها باعث ایجاد تاخیر در سیستم شده و نتیجه این تاخیر، کاهش کارایی سیستم می-

باشد. بنابراین در این مقاله برای شبیه‌سازی روش LFNC از ۵ گره مه در هر کلونی استفاده شده است.

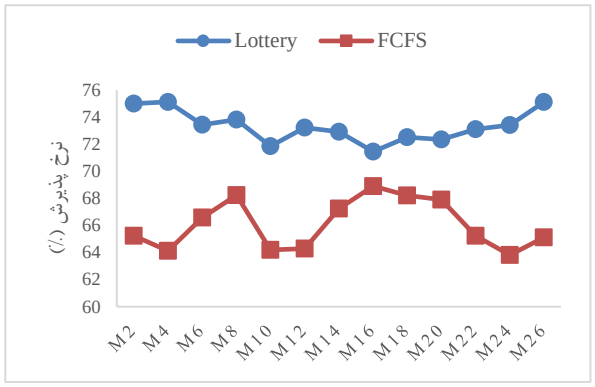
۴-۳- ارزیابی کارایی روش پیشنهادی

در آزمایش اول به بررسی کارایی زمانبندی درخواست‌های مهلت‌دار با استفاده از الگوریتم بخت‌آزمایی و بدون در نظر گرفتن همکاری گره‌های مه می‌پردازیم. در این آزمایش درخواست‌ها به گره‌های مه ارسال می‌شوند و اگر گره مه قادر به پاسخگویی به درخواست نباشد و همچنین مهلت زمانی درخواست نقض نشود، درخواست برای اجرا به ابر فرستاده می‌شود. برای زمانبندی درخواست‌ها یک‌بار از الگوریتم بخت‌آزمایی و بار دیگر از الگوریتم پایه FCFS در مؤلفه زمانبند درخواست در گره‌های مه استفاده شده است.

در شکل‌های ۵ و ۶ کارایی الگوریتم‌های بخت‌آزمایی (Lottery) و FCFS بر اساس معیارهای زمان پاسخ و نرخ پذیرش درخواست‌ها مقایسه شده است. همان‌طور که مشاهده می‌شود، زمان پاسخ در حالتی که از الگوریتم بخت‌آزمایی برای زمانبندی استفاده می‌شود در مقایسه با الگوریتم پایه FCFS کاهش و نرخ پذیرش درخواست-ها افزایش یافته است. به صورت دقیق‌تر، در پایان بازه در نظر گرفته شده برای شبیه‌سازی، زمان پاسخ الگوریتم بخت‌آزمایی در مقایسه با FCFS به میزان ۷.۱۳ میلی‌ثانیه و نرخ پذیرش درخواست‌ها به میزان ۹.۹۸ درصد بهبود یافته است. دلیل این امر اولویت‌دهی به درخواست‌های حساس به تاخیر است که اجرای دیرهنگام آن موجب نقض مهلت زمانی و عدم پذیرش درخواست می‌شود. این در حالی است که الگوریتم FCFS به مهلت زمانی درخواست‌ها توجه ندارد و ممکن است چند درخواست حساس به تاخیر در انتهای صف و بعد از یک درخواست غیر حساس به تاخیر با زمان اجرای بالا قرار گیرد.



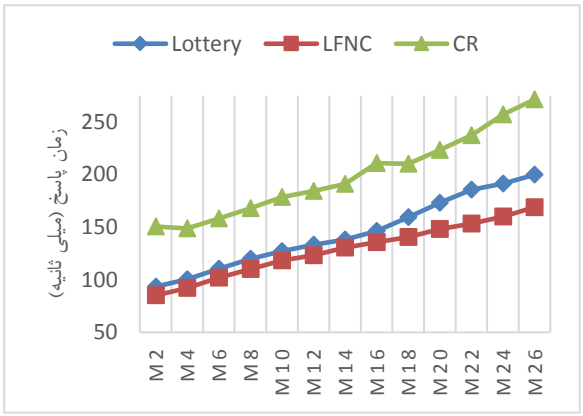
شکل ۵: مقایسه زمان پاسخ روش‌های Lottery و FCFS.



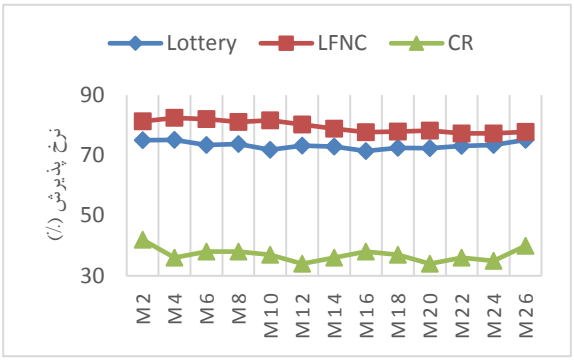
شکل ۶: مقایسه نرخ پذیرش روش‌های Lottery و FCFS.

در آزمایش دوم به منظور بررسی تاثیر همکاری گره‌های مه برای جایابی وظایف، سه روش در نظر گرفته شده است: ۱) $CR^{۷}$: همه درخواست‌ها به ابر ارسال شده و در آنجا اجرا شوند، ۲) Lottery: درخواست‌های کاربر به نزدیکترین گره مه ارسال می‌شوند و اگر گره مه قادر به پاسخگویی به درخواست نباشد و همچنین مهلت زمانی درخواست نقض نشود، درخواست برای اجرا به ابر فرستاده می‌شود ولی اگر گره مه قادر به پاسخگویی به درخواست باشد، درخواست‌های دریافتی با توجه به الگوریتم بخت‌آزمایی زمانبندی می‌شوند و ۳) LFNC: درخواست‌ها به نزدیک‌ترین گره مه فرستاده می‌شوند، گره مه جاری بر اساس روش پیشنهادی درباره مکان مناسب برای پردازش درخواست که می‌تواند خودش، یکی از گره-های همسایه یا ابر باشد تصمیم می‌گیرد.

در شکل‌های ۷ و ۸ به ترتیب زمان پاسخ و نرخ پذیرش سه روش فوق مقایسه شده است. همان‌طور که در شکل‌های ۷ و ۸ مشاهده می‌شود روش CR به دلیل فاصله زیاد مراکز داده ابر از کاربران و گلوگاه شدن پهنای باند شبکه، در پاسخ‌گویی به درخواست‌های مهلت‌آگاه، کارایی لازم را ندارد. زمان پاسخ و نرخ پذیرش در روش CR به ترتیب برابر ۲۷۱.۳۴ میلی‌ثانیه و ۴۰ درصد است که نشان می‌دهد ابر در پاسخگویی به درخواست‌های حساس به تاخیر عملکرد مناسبی ندارد. در واقع، ارسال کلیه‌ی درخواست‌ها به ابر موجب تحمیل زمان انتقال و انتشار بالا به این درخواست‌ها شده و منجر به ازدحام درخواست‌ها در ابر می‌شود. به عبارت دیگر ماشین مجازی موجود در ابر در حین اجرای همزمان درخواست‌های همگن، به هر درخواست منابع محاسباتی کمتری اختصاص داده و در نتیجه زمان اجرا بالا می‌رود. در مقابل بکارگیری معماری سه لایه و همکاری ابر-مه منجر به بهبود چشمگیری در معیارهای کارایی شده است.



شکل ۷: مقایسه زمان پاسخ روش‌های CR، Lottery و LFNC.

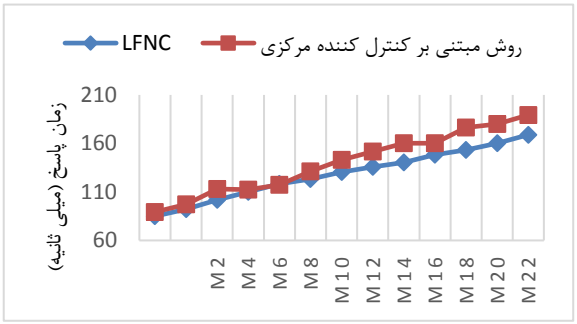


شکل ۸: مقایسه نرخ پذیرش روش‌های CR، Lottery و LFNC.

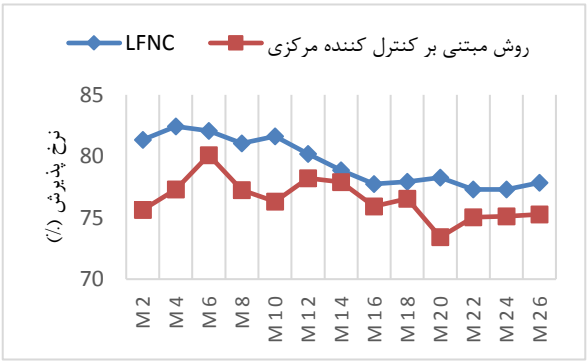
مطابق شکل‌های ۷ و ۸، زمان پاسخ و نرخ پذیرش الگوریتم بخت-آزمایی مبتنی بر همکاری ابر-مه به ترتیب ۱۹۹.۷۶ میلی‌ثانیه و ۷۵.۱ درصد است که تاثیر بکارگیری معماری سلسله‌مراتبی رایانش مه در پاسخگویی به درخواست‌های مهلت‌آگاه را نشان می-دهد. به بیانی دیگر، معماری مه با فراهم‌سازی فضای ذخیره‌سازی و پردازشی در لبه شبکه و در نزدیکی کاربر منجر به کاهش تاخیر در پاسخ به درخواست‌ها و به تبع آن کاهش درصد درخواست‌های رد شده به دلیل نقض مهلت زمانی می‌شود.

همچنین شکل‌های ۷ و ۸ نشان می‌دهند که استفاده از همکاری گره‌های مه در زمانبندی بخت‌آزمایی (روش LFNC) منجر به بهبود معیارهای زمان پاسخ و نرخ پذیرش شده است. در مقایسه با الگوریتم بخت‌آزمایی بدون همکاری گره‌های مه، معیار زمان‌پاسخ به میزان ۳۰.۸۴ میلی‌ثانیه کاهش یافته و معیار نرخ پذیرش ۲.۷۴ درصد افزایش یافته است. همکاری گره‌های مه برای اجرای درخواست‌ها باعث می‌شود تعداد قابل توجهی از درخواست‌ها که در صورت انتظار در صف گره مه جاری یا ارسال به ابر مهلت زمانی آنها نقض می‌شود بتوانند توسط گره‌های مه همسایه که تعداد درخواست‌های کمتری برای اجرا دارند در مهلت تعیین شده اجرا شوند.

در آزمایش سوم کارایی روش پیشنهادی (LFNC) با روش مبتنی بر کنترل‌کننده مرکزی برای همکاری گره‌های مه [۱، ۲] مقایسه شده است. در این روش هر کلونی مه برای اجرای سرویس و مدیریت منابع خود، دارای یک گره کنترل‌کننده مرکزی است. ارتباط یک کلونی با دیگر کلونی‌ها و همچنین ارتباط کلونی با ابر از طریق این گره صورت می‌پذیرد. همان‌گونه که در شکل‌های ۹ و ۱۰ مشاهده می‌شود، زمان پاسخ درخواست‌های کاربران در حالتی که از گره کنترل‌کننده مرکزی استفاده شده، به میزان ۲۰.۴۲ میلی‌ثانیه افزایش و نرخ پذیرش درخواست‌ها به میزان ۲.۵۷ درصد کاهش یافته است. دلیل این امر گلوگاه شدن گره کنترل‌کننده مرکزی است زیرا تصمیم‌گیری در رابطه با محل پردازش کلیه درخواست‌ها در این گره انجام می‌شود. به همین دلیل برای گره کنترل‌کننده یک صف ایجاد می‌شود. درخواست‌های منتظر در صف متحمل تاخیر شده و این تاخیر به وجودآمده باعث افزایش زمان پاسخ درخواست‌ها و نقض مهلت زمانی برخی از درخواست‌ها می‌شود.



شکل ۹: مقایسه زمان پاسخ روش پیشنهادی (LFNC) با روش مبتنی بر کنترل کننده مرکزی [۱، ۲].



شکل ۱۰: مقایسه نرخ پذیرش روش پیشنهادی (LFNC) با روش مبتنی بر کنترل‌کننده مرکزی [۱، ۲].

۵- نتیجه‌گیری

در این مقاله روشی کارا برای زمانبندی وظایف مهلت‌آگاه در رایانش مه با هدف کاهش زمان پاسخ و افزایش نرخ پذیرش درخواست‌های کاربران معرفی شد. روش پیشنهادی برای جایابی وظایف مبتنی بر همکاری گره‌های مه است و از الگوریتم بخت-آزمایی برای زمانبندی مبتنی بر اولویت درخواست‌ها بهره می-جوید. همکاری بین گره‌ها به این صورت است که زمان اتمام کار درخواست ارسالی به کلونی در صورت اجرا در هر یک از گره‌های کلونی یا ابر تخمین زده می‌شود و درخواست به گرهی ارسال می-شود که کمترین تخمین زمان اتمام کار را دارا باشد. همچنین در الگوریتم بخت‌آزمایی با توجه به مهلت زمانی درخواست‌ها به آنها اولویت داده می‌شود. بدین ترتیب تعداد درخواست‌های حساس به تاخیری که به دلیل نقض مهلت زمانی رد می‌شوند کاهش می‌یابد.

نتایج آزمایشات انجام گرفته نشان می‌دهد که الگوریتم بخت-آزمایی برای زمانبندی وظایف مهلت‌آگاه در مقایسه با الگوریتم پایه خدمت به ترتیب ورود کاراتر است و معیارهای نرخ پذیرش و زمان پاسخ را به ترتیب به میزان ۹.۹۸ درصد و ۷.۱۳ میلی‌ثانیه بهبود داده است. همچنین همکاری گره‌های مه منجر به بهبود معیارهای زمان پاسخ و نرخ پذیرش الگوریتم بخت‌آزمایی می‌شود. در روش پیشنهادی در مقایسه با الگوریتم بخت‌آزمایی بدون همکاری گره‌های مه، معیار زمان پاسخ ۳۰.۸۴ میلی‌ثانیه کاهش و معیار نرخ پذیرش ۲.۷۴ افزایش داشته است. به علاوه، روش غیرمتمرکز پیشنهادی برای همکاری گره‌های مه از روش‌های دارای گره کنترل‌کننده مرکزی عملکرد بهتری دارد و زمان پاسخ و نرخ پذیرش را به ترتیب به میزان ۲۰.۴۲ میلی‌ثانیه و ۲.۵۷ درصد بهبود داده است.

برخی از محدودیت‌های روش پیشنهادی که می‌توانند به عنوان کار آتی مدنظر قرار گیرند عبارتند از: ۱) اجرای ناموفق یک درخواست (مثلا به دلیل خرابی پردازنده) و نیاز به ارسال درخواست برای گره مه دیگر که خود باعث افزایش زمان پاسخ می‌شود، ۲) در نظر گرفتن درخواست کاربر به عنوان کوچکترین واحد پردازشی در روش پیشنهادی، در حالی که در اغلب کاربردها می‌توان یک درخواست را به تعدادی وظیفه کوچکتر و مستقل شکست که امکان اجرای موازی درخواست بر روی چند گره مه را فراهم می-کند، ۳) در نظر گرفتن معیارهای نرخ پذیرش و زمان پاسخ برای جایابی و زمانبندی وظایف در حالی که می‌توان سایر معیارها مانند مصرف انرژی و امنیت را نیز بررسی کرد، ۴) در نظر گرفتن شبکه به صورت ایستا در حالی که کاربران و گره‌های مه می‌توانند متحرک باشند. تحرک کاربران در کنار نیازمندی‌های کیفیت سرویس متفاوت آن‌ها، ارائه روش‌های جدید جایابی و زمانبندی وظایف را که بتوانند کارایی مطلوب را برای کاربران متحرک فراهم کنند، ضروری می‌سازد. از سوی دیگر تحرک بالای کاربران موبایل در میان گره‌های مه با ناحیه دسترسی محدود، در برخی موارد می‌تواند موجب از دست رفتن یک درخواست شود و یا باعث گردد که درخواست در مرکز داده با فاصله‌ی جغرافیایی دور نسبت به کاربران نهایی پردازش شود. این امر به خصوص در برنامه‌های کاربردی حساس به تاخیر، موجب افزایش تاخیر و رد درخواست‌ها می‌شود. به عنوان کار آتی پیشنهاد می‌شود تحرک کاربران بین گره‌های مه در روش پیشنهادی برای زمانبندی وظایف در نظر گرفته شود.

مراجع

[1] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski and P. Leitner, "Optimized IoT service placement in the fog," Service Oriented Computing and Applications, Vol. 11, No. 4, pp. 427-443, 2017.

[2] O. Skarlat and S. Schulte, "FogFrame: a framework for IoT application execution in the fog," PeerJ Computer Science, Vol. 7, pp. e588, 2021.

[3] J. Nie, J. Luo and L. Yin, "Energy-aware Multi-dimensional Resource Allocation Algorithm in Cloud Data Center", Ksii Transactions on Internet & Information Systems, Vol. 11, No. 9, 2017.

[4] EH. Houssein, AG. Gad, YM. Wazery PN.and Suganthan, "Task scheduling in cloud computing based on meta-heuristics: review, taxonomy, open challenges, and future trends," Swarm and Evolutionary Computation, Vol. 62, pp.100841, 2021.

[5] PK. Bal, SK. Mohapatra, TK. Das, K. Srinivasan and YC. Hu, "A Joint Resource Allocation, Security with Efficient Task Scheduling in Cloud Computing Using Hybrid Machine Learning Techniques," Sensors, Vol. 22, No. 3, pp. 1242, 2022.

[6] AM. Yadav, KN. Tripathi and SC Sharma, "An enhanced multi-objective fireworks algorithm for task scheduling in fog computing environment," Cluster Computing, Vol. 25, No. 2, pp. 983-98, 2022.

[7] J. Gu, J. Mo, B. Li, Y. Zhang and W. Wang, "A multi-objective fog computing task scheduling strategy based on ant colony algorithm," In 2021 IEEE 4th International Conference on Information Systems and Computer Aided Education (ICISCAE), pp. 12-16, 2021.

[8] N. Potu, C. Jatoth, and P. Parvataneni, "Optimizing resource scheduling based on extended particle swarm optimization in fog computing environments,"Concurrency and Computation: Practice and Experience, Vol. 33, No. 23, pp. e6163, 2021.

[9] K.P.N. Jayasena and B.S. Thisarasinghe, "Optimized task scheduling on fog computing environment using meta heuristic algorithms," In 2019 IEEE International Conference on Smart Cloud (SmartCloud), pp. 53-58, 2019.

[10] M. Abd Elaziz, L. Abualigah and I. Attiya, "Advanced optimization technique for scheduling IoT tasks in cloud-fog computing environments," Future Generation Computer Systems, 2021.

[11] M. Verma, N. Bhardwaj and A.K. Yadav, "Real Time Efficient Scheduling Algorithm for Load Balancing in Fog Computing

Environment," I.J. Information Technology and Computer Science," Vol. 8, No. 4, pp. 1-10, 2016.

[12] T. Choudhari, M. Moh and T.S. Moh, "Prioritized task scheduling in fog computing," In Proceedings of the ACMSE 2018 Conference, pp. 1-8, 2018.

[13] Y. Xiao and M. Krunz, "QoE and power efficiency tradeoff for fog computing networks with fog node cooperation," IEEE INFOCOM 2017-IEEE Conference on Computer Communications, pp. 1-9, 2017.

[14] S. Azizi, F. Khosroabadi and M. Shojafar, "A priority-based service placement policy for fog-cloud computing systems," Computational Methods for Differential Equations, No. 4 (Special Issue), pp. 521-34, 2019.

[15] C. Guerrero, I. Lera, C. Juiz, "On the influence of fog colonies partitioning in fog application makespan," In IEEE International Conference on Future Internet of Things and Cloud (FiCloud), pp. 377-384, 2018.

[16] P. Maiti, B. Sahoo, A.K. Turuk, A. Kumar and B.J. Choi, "Internet of Things applications placement to minimize latency in multi-tier fog computing framework," ICT Express, 2021.

[17] S. Wang, T. Zhao and S. Pang, "Task scheduling algorithm based on improved firework algorithm in fog computing," IEEE Access, 8, pp. 32385-32394, 2020.

[18] O. Consortium, "OpenFog reference architecture for fog computing," Architecture Working Group, pp. 1-162, 2017.

[19] RN. Calheiros, R. Ranjan, A. Beloglazov, CA. De Rose and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Software: Practice and experience, Vol. 41, No. 1, pp. 23-50, 2011.

[20] H.E. Refaat and M.A. Mead, "DLBS: Decentralize Load-Balance Scheduling Algorithm for Real-Time IoT Services in Mist Computing," Editorial Preface from the Desk of Managing Editor, Vol. 10, No. 9, 2019.

[21] A. Khalid and M. Shahbaz, "Service Architecture Models For Fog Computing: A Remedy for Latency Issues in Data Access from Clouds," TIIS, Vol. 11, No. 5, pp. 2310-2345, 2017.

پاورقی‌ها:

[1] Fog computing

[2] Real-Time Efficient Scheduling

[3] IP Network

[4] CloudSim

[5] Accept Ratio over Response Time

[6] Lottery Fog Node Collaboration

[7] Cloud Response (CR)