



**مؤسسه آموزش عالی سلمان**

**گروه مهندسی‌کامپیوتر**

**پایان‌نامه کارشناسی‌ارشد (M. Sc.)**

**در رشته‌ی مهندسی فناوری اطلاعات-شبکه‌های کامپیوتری**

**عنوان:**

**زمانبندی در محیط‌های محاسباتی لبه و مه مبتنی بر یادگیری عمیق با استفاده از شبکه عصبی حافظه طولانی کوتاه مدت برای بهینه‌سازی بار سیستم و زمان پاسخ**

**استاد راهنما:**

**دکتر جواد حمیدزاده**

**نگارش:**

**علی شم آبادی**

**زمستان 1403**

**تقدیم به سه وجود مقدس  
آنان که ناتوان شدند تا ما به توانایی برسیم... موهایشان سپید شد تا ما روسفید شویم... و عاشقانه سوختند تا گرمابخش وجود ما و روشنگر راهمان باشند...**

**پدرانمان**

**مادرانمان**

**استادانمان**

**تقدیر**

**از اساتید بزرگوارم برای تمام حمایت‌ها و زحمات بی دریغ‌شان سپاس‌گزاری می‌کنم. از استاد راهنمایم که زحمت راهنمایی این رساله را به عهده داشتند سپاس فراوان دارم.. از دوستان و همکلاسی‌های گرامی خود که ایام خوشی را در کنار هم سپری کردیم و مرا در رسیدن به اهدافم یاری نمودند صمیمانه تقدیر می‌کنم.. در پایان از کلیه دوستان خوبم که مطالب زیادی را به من آموختند تشکر کرده و برایشان آرزوی موفقیت و سربلندی می‌کنم.**

**چکیده**

**کلمات کلیدی:** محاسبات لبه، محاسبات مه، زمانبندی، یادگیری عمیق، شبکه عصبی LSTM تعادل بار، زمان پاسخ.

**فهرست مطالب**

**عنوان صفحه**

[**فصل اول) کلیات پژوهش** 1](#_Toc166318049)

[1-1) مقدمه 2](#_Toc194838807)

[2-1) بیان مسئله 2](#_Toc194838808)

[3-1) اهمیت و ضرورت تحقیق 6](#_Toc194838809)

[1-3-1) اهمیت تحقیق 8](#_Toc194838810)

[2-3-1) ضرورت تحقیق 9](#_Toc194838811)

[4-1) جنبه نوآوری پژوهش 10](#_Toc194838812)

[5-1) اهداف پژوهش 11](#_Toc194838813)

[6-1) سؤالات پژوهش 11](#_Toc194838814)

[7-1) فرضیات پژوهش 12](#_Toc194838815)

[8-1) توصیف مفاهیم اصلی پژوهش 12](#_Toc194838816)

[9-1) ساختار پایان‌نامه 13](#_Toc194838817)

[**فصل دوم) پیشینه پژوهش** 15](#_Toc166318049)

[1-2) مقدمه 16](#_Toc194838818)

[2-2) ادبیات پژوهش 16](#_Toc194838819)

[1-2-2) فرایندهای زمانبندی در محاسبات مه 17](#_Toc194838820)

[2-2-2) چالش‌های محاسبات مه و لبه 22](#_Toc194838821)

[3-2-2) بارگذاری وظایف در محاسبات لبه و مه 25](#_Toc194838822)

[3-2) پیشینه پژوهش 28](#_Toc194838823)

[1-3-2) زمان‌بندی کار عظیم انرژی و تأخیری در سیستم محاسباتی لبه و مه 48](#_Toc194838824)

[2-3-2) زمان‌بندی وظایف در سیستم‌های محاسباتی مه ابری 50](#_Toc194838825)

[3-3-2) زمان‌بندی وظایف چند هدفه در محاسبات ابری و مه با استفاده از رویکرد برنامه نویسی هدف 53](#_Toc194838826)

[4-3-2) مشخص کردن زمان‌بندی برنامه ها در منابع محاسباتی لبه، مه و ابر 56](#_Toc194838827)

[5-3-2) زمان‌بندی کار آگاه از هزینه در محیط مه-ابر 56](#_Toc194838828)

[4-2) مقایسه روش‌های موجود 60](#_Toc194838829)

[5-2) خلاصه و نتیجه‌گیری 62](#_Toc194838830)

[**فصل سوم) روش پیشنهادی** 63](#_Toc166318049)

[1-3) مقدمه 64](#_Toc194838831)

[2-3) روش شناسی پژوهش 65](#_Toc194838832)

[3-3) مدل مفهومی پژوهش 65](#_Toc194838833)

[4-3) الگوریتم‌ها و راهکارها 67](#_Toc194838834)

[1-4-3) پارامترهای کیفیت سرویس 67](#_Toc194838835)

[2-4-3) پیش پردازش اطلاعات 72](#_Toc194838836)

[3-4-3) نرمال‌سازی اطلاعات 75](#_Toc194838837)

[4-4-3) شبکه عصبی LSTM 75](#_Toc194838838)

[5-3) زمانبندی وظایف با استفاده از روش پیشنهادی 83](#_Toc194838839)

[6-3) فلوچارت روش پیشنهادی 83](#_Toc194838840)

[7-3) الگوریتم نهایی 85](#_Toc194838841)

[8-3) جمع بندی فصل 85](#_Toc194838842)

[**فصل چهارم) ارزیابی نتایج** 87](#_Toc166318049)

[1-4) مقدمه 88](#_Toc194838843)

[2-4) معرفی محیط شبیه سازی 88](#_Toc194838844)

[3-4) متغیرهای شبیه سازی 90](#_Toc194838845)

[4-4) معیارهای عملکرد در فرایند شبیه سازی 92](#_Toc194838846)

[5-4) نتایج شبیه سازی 92](#_Toc194838847)

[6-4) ارزیابی نتایج 99](#_Toc194838848)

[**فصل پنجم) جمع بندی و پیشنهادات آتی** 100](#_Toc166318049)

[1-5) جمع بندی 101](#_Toc194838850)

[2-5) پیشنهادات آتی 104](#_Toc194838851)

[**منابع و مراجع 106**](#_Toc194838852)

**فهرست شکل‌ها**

**عنوان صفحه**

[شکل 2- 1) چارچوب معماری محاسبات مه-لبه با زمان‌بندی وظایف[1] 19](#_Toc185437634)

[شکل 2- 2) مدل پیشنهادی گروه تحقیقاتی نجف زاده و همکارانش[40] 54](#_Toc185437635)

[شکل 2- 3) مدل پیشنهادی گروه تحقیقاتی نیکویی و همکارانش[16] 58](#_Toc185437636)

[شکل 3- 1) مدل مفهومی رویکرد پیشنهادی 65](#_Toc189493374)

[شکل 3- 2) مدل آموزش در شبکه عصبی 75](#_Toc189493375)

[شکل 3- 3) معماری اولویت‌بندی پارامترهای کیفیت سرویس مبتنی بر شبکه عصبی LSTM 78](#_Toc189493376)

[شکل 3- 4) ساختار گیت فراموشی مبتنی بر شبکه عصبی LSTM 79](#_Toc189493377)

[شکل 3- 5) ساختار گیت ورودی مبتنی بر شبکه عصبی LSTM 80](#_Toc189493378)

[شکل 3- 6) ساختار گیت خروجی مبتنی بر شبکه عصبی LSTM 81](#_Toc189493379)

[شکل 3- 7) فلوچارت رویکرد پیشنهادی 83](#_Toc189493380)

[شکل 4- 1) مقایسه تأخیر میانگین روش پیشهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] 93](#_Toc194838857)

[شکل 4- 2) مقایسه سربار محاسباتی روش پیشهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] 94](#_Toc194838858)

[شکل 4- 3) مقایسه تعادل بار روش پیشهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] 95](#_Toc194838859)

[شکل 4- 4) مقایسه انرژی مصرفی روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] 96](#_Toc194838860)

[شکل 4- 5) مقایسه نرخ همگرایی به جواب بهینه برای روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] 97](#_Toc194838861)

[شکل 4- 6) مقایسه زمان پاسخ برای روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] 98](#_Toc194838862)

**فهرست جداول**

**عنوان صفحه**

[جدول 2- 1) جدول مقایسه روش‌های موجود 59](#_Toc166317944)

[جدول 4- 1) متغیرهای شبیه سازی 89](#_Toc194838921)

فصل اول

**کلیات تحقیق**

**اهداف فصل:**

* توصیف محیط محاسبات لبه و مه
* معرفی چالش‌های مهم در محاسبات لبه و مه
* معرفی مسئله زمانبندی در محاسبات لبه و مه محور و اهمیت آن
* معرفی ساختار پایان‌نامه

# 1-1) مقدمه

این فصل به بررسی شبکه های محاسباتی لبه و مه، نحوه ایجاد و هدف از ایجاد آنها می پردازد. سپس چالش های اساسی شبکه های محاسباتی لبه و مه در دستیابی به بهترین عملکرد ارائه شده است. در ادامه چالشی با عنوان زمانبندی در محیط محاسباتی لبه و مه به عنوان چالش مورد مطالعه در نظر گرفته شده و ساختار آن بررسی خواهد شد. همچنین با توصیف کلی یک راهکار جدید در این زمینه، نوآوری پژوهش بیان شده و در ادامه اهداف و فرضیات اساسی در این پژوهش بیان خواهد شد. در نهایت مفاهیم اصلی و متغیرهای پژوهش بیان شده و سپس ساختار پایان نامه توصیف می شود.

# 2-1) بیان مسئله

در چند سال گذشته شاهد رشد سریع صنعت اینترنت اشیا [[1]](#footnote-1) بوده ایم که امکان اتصال مردم به اشیا و ساختارهای هوشمند فیزیکی را فراهم کرده و روند و دیجیتالی شدن دنیای فیزیکی را تسهیل می کند. در همین حال، با رشد دستگاه های اینترنت اشیا و کاربردهای مختلف، انتظار برای پایداری و تأخیر پایین تر این ساختارها بیشتر از همیشه شده است. رایانش ابری، به عنوان فعال کننده اصلی اینترنت اشیا شناخته شده است و داده ها و اطلاعات تولید شده توسط دستگاه های اینترنت اشیا را ذخیره و پردازش می کند. رایانش ابری با استفاده از قابلیت های محاسباتی قدرتمند و فناوری های ذخیره سازی پیشرفته، امنیت و قابلیت اطمینان اطلاعات ذخیره شده را تضمین می کند. با این حال، سرورها در ساختار رایانش ابری معمولاً در فاصله فیزیکی طولانی از دستگاه های اینترنت اشیا قرار دارند و تأخیر زیاد ناشی از فواصل طولانی نمی تواند به طور مؤثر نیازهای مربوط به برنامه های اینترنت اشیا با نیازهای پردازش آنی را برآورده کند[1]. به دلیل این مسائل، محاسبات لبه و مه به عنوان یک ساختار محاسباتی محبوب در زمینه اینترنت اشیا پدیدار شده اند. اگرچه برخی از محققان از اصطلاحات محاسبات لبه و محاسبات مه به جای یکدیگر استفاده می کنند، اما به طور کلی، ساختاری که از منابع لبه برای برنامه های کاربردی اینترنت اشیا به صورت پردازش بلادرنگ استفاده می کند، به عنوان محاسبات لبه در نظر گرفته شده و ساختاری که از لبه استفاده می کند و هر زمان که لازم باشد نیز از منابع ابری استفاده کند، به عنوان محاسبات مه در نظر گرفته شده است. محاسبات لبه به عنوان یک معماری محاسباتی غیرمتمرکز، پردازش، ذخیره سازی و کنترل هوشمند را در مجاورت دستگاه های اینترنت اشیا به ارمغان می آورد. این معماری انعطاف پذیر، خدمات محاسباتی ابری را تا لبه شبکه گسترش می دهد. در مقابل، پارادایم محاسباتی مه، از مزایای محاسبات ابری و لبه برخوردار است که نه تنها قابلیت های محاسباتی قدرتمندی را فراهم می کند، بلکه نیاز به انتقال داده ها به ابر برای پردازش، تجزیه و تحلیل و ذخیره سازی را کاهش می دهد، در نتیجه نرخ از بین رفتن بین شبکه را کاهش می دهد[2].

محاسبات لبه و مه پشتیبان قدرتمندی را برای نوآوری و توسعه در زمینه های مختلف فراهم می کند. به عنوان مثال، در زمینه مراقبت های بهداشتی هوشمند، استقرار گره های محاسباتی لبه بر روی دستگاه های پزشکی می تواند پارامترهای فیزیولوژیکی بیماران را در زمان واقعی پایش کرده و داده ها را برای تجزیه و تحلیل و تشخیص، تحقق پزشکی از راه دور و پزشکی شخصی سازی شده به ابر منتقل کند. با این حال، رشد گسترده در تعداد اینترنت اشیا، برنامه ها و سرورها در محیط های محاسباتی مه نیز چالش های جدیدی ایجاد می کنند. در مرحله اول، انتظار می رود زمان اجرا به حداقل برسد، به این معنی که برنامه ها باید توسط بهترین سرور (یعنی قوی ترین و از نظر فیزیکی نزدیک ترین سرور) پردازش شوند. علاوه بر این، بار سیستم باید به طور ایده آل متعادل سازی و توزیع شود تا بر روی چندین واحد عملیاتی اجرا شود[1]. به عنوان مثال، با توزیع درخواست ها در چندین سرور به روشی یکپارچه (مانند محیط های محاسباتی بدون سرور)، تعادل بار می تواند از بارگذاری بیش از حد سرورهای موجود در شبکه جلوگیری کند و اطمینان حاصل کند که هر سرور بار پردازشی متوسطی را مدیریت می کند. این امر زمان پاسخ، عملکرد کلی سیستم و توان عملیاتی را بهبود می بخشد و همچنین به سرورها کمک می کند تا با ثبات تر کار کنند. بنابراین، بهبود سطح توازن بار سرورها (به عنوان مثال، کاهش حجم استفاده از منابع سرور) و در عین حال کاهش زمان پاسخ به یک مسئله مهم اما چالش برانگیز برای زمان بندی برنامه های اینترنت اشیا در سرورها در محیط های محاسبه لبه / مه تبدیل شده است[3].

چارچوب محاسبات لبه / مه امکان نظارت، تجزیه و تحلیل، پردازش، تجزیه و تحلیل، نگهداری، توزیع و کنترل محاسبات را در نزدیکی گره های اینترنت اشیا فراهم نموده و همچنین امکانات ذخیره سازی، ارتباطات و تصمیم گیری را برای کاربران نهایی ایجاد می کند. پیچیدگی محاسبات لبه / مه منجر به چالش در تکمیل وظایف در یک چارچوب زمانی معین به دلیل حجم زیادی از داده هایی که باید پردازش شوند، می شود. بنابراین، زمان بندی وظایف نقش بسیار مهمی دارد. الگوریتم های زمان بندی با چالش زیادی برای مدیریت چنین وظایفی روبه رو هستند، زیرا تأخیر در چنین وظایفی غیرقابل قبول است. یک رویکرد زمانبندی وظایف مسئول زمانبندی منابع، مانند CPU و حافظه، برای یک کار خاص است. معمولاً مشکلات بهینه سازی محاسبات لبه / مه برای زمان بندی وظایف، با استفاده از توزیع منابع، ذخیره سازی داده ها و قرار دادن کپی ها فرموله می شوند. هدف یک الگوریتم بهینه سازی یافتن بهترین راه حل از بین چندین احتمال است. با این حال، مشکل زمانبندی وظایف می تواند یک چالش حیاتی برای استقرار محاسبات لبه / مه، علی رغم مزایای قابل توجه آن نسبت به رایانش ابری باشد. اگر سرورها و منابع مناسب به آنها تخصیص داده نشود، پردازش وظایف می تواند سنگین باشد. ارائه دهندگان خدمات ممکن است در نتیجه این امر جریمه های شدیدی را متحمل شوند. سناریوهای اینترنت اشیا با تعداد زیادی دستگاه موجود، عدم تعادل در استفاده از منابع را نشان می دهند. بنابراین، تکنیک های زمان بندی بلادرنگ که تأخیرهای سرویس را کاهش می دهند، از نظر انرژی مقرون به صرفه هستند و می توانند تأخیر سرویس را کاهش دهند، برای شبکه های لبه / مه ضروری هستند. جدای از این، وظایف اینترنت اشیا در محاسبات و ذخیره سازی (به عنوان مثال، CPU، حافظه و مهلت زمانی) تحت محدودیت های مختلفی قرار دارند. علاوه بر این، زمانبندی منابع اینترنت اشیا در منابع ابری یا منابع لبه / مه باید چندین پارامتر کیفیت سرویس[[2]](#footnote-2) (به عنوان مثال، تأخیر، انرژی) را در نظر گرفته و از سویی دیگر هم کاربران نهایی و هم طراحان سیستم باید این پارامترها را هنگام طراحی برنامه های اینترنت اشیا در نظر بگیرند[4]. وظایف در محیط مه به دو دسته، داده فشرده یا محاسبات فشرده طبقه بندی می شوند. در طول زمان بندی، از منابع محاسباتی با کارایی بالا برای محاسبات فشرده و برای انتقال داده ها از منابع با کارایی پایین استفاده می شود که منجر به کاهش زمان اجرای کار می شود. تعداد انتقال داده ها در طول زمان بندی وظایفی که به شدت داده نیاز دارند به حداقل می رسد. این باعث کاهش زمان انتقال داده می شود. با توجه به اینکه امروزه تعداد دستگاه های اینترنت اشیا در حال افزایش است، این روند باعث می شود لبه / مه و گره های ابری با وظایف پردازشی بیشتری مواجه شوند. بنابراین، برای زمانبندی وظایف و مدیریت منابع در محیط های لبه / مه، یک تکنیک کارآمد مورد نیاز است[1].

از آنجایی که زمابندی، یک چالش از نوع NP Hard[[3]](#footnote-3) است، می توان راه حل های فراابتکاری را به منظور مدیریت مسائل زمانبندی در نظر گرفت. با این حال، این رویکردها اغلب بر دانش کلب از اطلاعات جهانی متکی هستند و پیشنهاد دهنده راه حل را ملزم به کنترل تغییرات می کنند. در محیط محاسباتی لبه / مه، اغلب هیچ نظمی در عملکرد سرور، استفاده و زمان خرابی وجود ندارد. تعداد برنامه های اینترنت اشیا و منابع مورد نیاز مربوطه تقریباً تصادفی است[5]. علاوه بر این، در واقعیت، به منظور مدیریت این ساختارها، از گراف های غیر چرخشی جهت دار برای مدل سازی برنامه های کاربردی اینترنت اشیا استفاده می شود، جایی که گره ها وظایف را نشان می دهند و یال های گراف ارتباط داده بین وظایف وابسته را نشان می دهند. وابستگی بین وظایف، پیچیدگی بالاتری را در زمان بندی برنامه ها ایجاد می کند. بنابراین، راه حل های فراابتکاری مبتنی بر قاعده نمی توانند به تنهایی و به طور مؤثر با چالش زمان بندی در اینترنت اشیا در محیط های محاسباتی لبه / مه روبه رو شوند[6].

در این راستا، محققان از ساختارهایی با عنوان یادگیری ماشین و یادگیری عمیق به منظور حل مسئله زمانبندی در محیط لبه و مه استفاده کرده اند، در یادگیری عمیق، عامل به طور مداوم با محیط در تعامل است و تعداد زیادی از مسیرهای تجربی (به عنوان مثال، توالی حالت ها، اقدامات و پاداش ها) را ثبت می کند که در مرحله آموزش برای یادگیری سیاست های بهینه استفاده می شود. برخلاف الگوریتم های فراابتکاری، عوامل یادگیری عمیق می توانند به طور مستقل تغییرات محیط را حس کنند و به آن واکنش نشان دهند، که به یادگیری عمیق اجازه می دهد تا مسائل پیچیده را در سناریوهای واقعی حل کند. با این حال، به دلیل محدودیت منابع محاسباتی دستگاه ها در محیط های محاسباتی ابر و مه، الزامات محاسباتی شبکه های عصبی عمیق اغلب پشتیبانی نمی شوند[7]. بنابراین، چگونگی ایجاد تعادل بین سادگی پیاده سازی، پیچیدگی نمونه و عملکرد راه حل، به یک چالش تحقیقاتی کلیدی در استفاده از یادگیری عمیق در محیط های محاسباتی لبه و مه در موقعیت های پیچیده تبدیل شده است[2]. برای مدیریت چالش های فوق، در این پژوهش، یک رویکرد زمانبندی کاربردی اینترنت اشیا مبتنی بر یادگیری عمیق پیشنهاد شد که از ساختار شبکه عصبی حافظه طولانی کوتاه مدت به منظور طبقه بندی منابع پردازشی و اجرای فرایند زمانبندی استفاده می کند. الگوریتم پیشنهادی، که از تکنیک طبقه بندی مبتنی بر یادگیری عمیق برای حل چالش زمانبندی برنامه های اینترنت اشیا در محیط های محاسباتی لبه و مه استفاده می کند. روش پیشنهادی می تواند به طور مؤثر هزینه متعادل سازی بار سرورها، هزینه زمان پاسخ برنامه های اینترنت اشیا و هزینه وزنی آنها را بهینه کند. علاوه بر این، با استفاده از روش پیشنهادی برای محدود کردن دفعات به روزرسانی پارامترهای کیفیت سرویس در هر تکرار، انتظار می رود سرعت همگرایی الگوریتم بهبود یابد.

# 3-1) اهمیت و ضرورت تحقیق

تکامل دستگاه های اینترنت اشیا منجر به تولید حجم عظیمی از داده ها شده است که نیاز به نظارت، پردازش و تجزیه و تحلیل دارند. به دلیل قدرت محاسباتی و حجم ذخیره سازی فراوان، ابر به مدلی شایسته برای اجرای برنامه های کاربردی کاربر تبدیل شده است. مرکز داده ابری برای پردازش داده های برنامه استفاده می شود که توسط ارائه دهندگان خدمات ابری مانند آمازون، گوگل یا مایکروسافت آژور مدیریت می شود. تأخیر قابل توجهی در پردازش برنامه ها در مرکز داده ابری به دلیل موقعیت جغرافیایی دور از کاربران مشاهده شده است. این تأخیر ممکن است برای برنامه هایی با زمان پاسخ دهی سخت گیرانه، مانند مراقبت های بهداشتی هوشمند غیرقابل قبول باشد. برای غلبه بر این محدودیت تأخیر، طراحان سیستم در حال بررسی محاسبات مه هستند. به طور کلی، یک معماری فقط ابری ممکن است برای برنامه های حساس به تأخیر مناسب نباشد، به دلیل مسافتی که در بالا مشخص شده است. محاسبات مه یک معماری متشکل از تعدادی دستگاه مه / مراکز داده میکرو، در مجاورت منابع تولید داده، مانند گوشی های هوشمند و حسگرها، ارائه می دهد. از این رو، داده های تولید شده را می توان در دستگاه های مه به موقع پردازش کرد[8], [9]. مشاهده شده است که دستگاه های مه دارای قدرت محاسباتی و ظرفیت ذخیره سازی محدودی هستند. معماری کلی محاسبات مه را می توان به سه لایه تقسیم کرد. اولین لایه، لایه دستگاه های IoT است که شامل انواع مختلف دستگاه ها، مانند گوشی های هوشمند، وسایل نقلیه هوشمند، رایانه های لوحی و دستگاه های مختلف خانه هوشمند است. این لایه می تواند محیط اطراف را حس کرده و از طریق دستگاه های حسگر داده ها را جمع آوری کند و از طریق فناوری های 3G، 6G، 5G، WiFi و بلوتوث با لایه محاسباتی مه ارتباط برقرار کند. لایه دوم لایه میانی، لایه محاسباتی مه است که شامل مسیریاب ها، دروازه ها، ایستگاه های کاری، سوئیچ ها، نقاط دسترسی است. این لایه قابلیت محاسبات، شبکه و ذخیره سازی را دارد. در نهایت، لایه بالایی محاسبات ابری است که شامل سرورهای ابری با قدرت محاسباتی بالا است. تغییراتی مانند دستگاه های کاربر از نظر پهنای باند، ذخیره سازی، تأخیر و محاسبات، مدیریت منابع در محیط محاسبات مه را به یک مسئله مهم تبدیل می کند. زمان بندی چالش اصلی در محاسبات مه است که ضرورت های خاص خود در این ساختارها را دارد، در ادامه به توصیف اهمیت و ضرورت پژوهش از جنبه های مختلف پرداخته می شود.

# 1-3-1) اهمیت تحقیق

در محیط مه، وظایف به دو گروه تقسیم می شوند: وظایفی که به شدت محاسبات نیاز دارند و وظایفی که به شدت داده نیاز دارند. در حالی که زمان بندی کارهایی که نیاز به شدت محاسبات دارند، زمان بندی داده ها را به منبع بهره وری بالا منتقل می کند و از این رو، زمان اجرای کار کاهش می یابد. از سوی دیگر، ضمن زمان بندی وظایفی که نیاز به شدت داده دارند، سعی می شود تعداد مهاجرت داده ها کاهش یابد. در نتیجه زمان انتقال داده کاهش می یابد. امروزه تعداد دستگاه های اینترنت اشیا در حال افزایش است که به نوبه خود بار روی گره های لبه و مه برای پردازش را افزایش می دهد. از این رو، به یک تکنیک کارآمد برای برنامه ریزی وظایف و مدیریت منابع محیط های مه و ابر نیاز دارد. علی رغم اهمیت رویکرد زمان بندی در محاسبات مه، هیچ نظرسنجی در مورد الگوریتم های زمان بندی در محیط مه وجود ندارد که به ضرورت های محقق در انجام وظایف زمان بندی کمک کند. به طور کلی مشکلات زمان بندی نیز به پنج دسته اصلی زمان بندی کار، زمان بندی منابع، تخصیص منابع، زمان بندی تخصیص کار، و زمان بندی گردش کار تقسیم شده اند. به طور کلی مشاهده می شود که ارائه یک رویکرد زمانبندی بهینه در فرایندهای محاسباتی مه اهمیت بالایی دارد.

هنگام حل مسئله **زمانبندی منابع** در محیط لبه، تعدادی از پارامترهای مهم مختلف ممکن است اعمال شود، زیرا سهام داران و بازیگران مختلف (به عنوان مثال، ارائه دهندگان ابر، ارائه دهندگان لبه، کاربران موبایل و ارائه دهندگان خدمات) اهداف مختلفی را هدف قرار می دهند. یک تابع هدف به فرمول بندی رسمی و ریاضی این اهداف و هدایت راه حل زمانبندی منابع در محیط لبه کمک می کند. به طور کلی، اهمیت به کارگیری ساختارهای زمانبندی منابع در محیط لبه و مه را می توان از زوایای متفاوت مانند کاهش تأخیر، بهبود مصرف انرژی، هزینه استقرار، بهبود دقت و پردازش چند هدفه بررسی کرد. به طور کلی در اجرای کار به حداقل رساندن تأخیر یکی از اهداف اصلی در طول مشکل زمانبندی منابع است. صرف نظر از نوع زمانبندی منابع، هدف کلی به کاهش کل تأخیر اجرای کار محدود می شود.

دومین هدف متداول در هنگام زمانبندی منابع، به حداقل رساندن مصرف انرژی است. مصرف انرژی معمولاً به دستگاه های پایانی اشاره دارد. دلیل آن این است که دستگاه های سیار و اینترنت اشیا معمولاً با باتری کار می کنند، بنابراین یک نگرانی عمده این است که چگونه می توان طول عمر باتری را با کاهش مصرف انرژی دستگاه به حداکثر رساند. همچنین مسئله زمانبدی اغلب به عنوان یک مسئله تخصیص منبع در نظر گرفته می شود که در آن منابع مناسب در ابر یا لبه باید با توجه به هزینه ابزار استقرار رزرو شوند تا وظایف بارگذاری شده در یک محیط مجازی اجرا شوند. این هزینه استقرار می تواند به طرق مختلفی باشد که هر کدام تفسیر متفاوتی دارند. یک هدف اضافی در فرایندهای زمانبندی منابع در محاسبات ابری، می تواند دقت مدل برای پیش بینی رفتار برنامه ها باشد. دقت مدل را می توان با استفاده از معیارهای مناسب بهینه کرد. به طور کلی مسئله افزایش دقت در این ساختارها به دلیل ارائه یک ساختار ارتباطی قابل اطمینان اهمیت قابل توجهی دارد.

# 2-3-1) ضرورت تحقیق

مزایای زمانبندی در محیط لبه و ابر بسیار زیاد است، زیرا این رویکردها امکان افزایش کیفیت سرویس در برنامه ها را فراهم می کنند و در عین حال عمر باتری دستگاه های پایانی را افزایش می دهند. برای دستیابی به آن، تأکید خاصی بر مدل های بهینه سازی مورد استفاده همراه با اهداف مختلف داده می شود تا وظایف دستگاه های پایانی را به یک منابع ارتباطی مناسب تقسیم و تخصیص دهد. از آنجایی که این مسئله شامل پارامترهای دینامیکی مختلفی از نظر شبکه و رفتار کاربر است، از این رو ضرورت بررسی و ارائه راهکارهای مناسب در این زمینه پیش از پیش نمایان می شود. از سویی دیگر اگرچه مسئله زمانبندی منابع در حوزه محاسبات ابری به خوبی مورد مطالعه قرار گرفته است، اما زمانبندی منابع در محیط هنوز با چالش های قابل توجهی در سناریوی محاسبات ابری مواجه است. به عنوان مثال، تأخیر اضافی انتقال و مصرف انرژی زمانی رخ می دهد که تعداد کارهای بارگذاری زیاد باشد، یا در سناریوهای پیچیده با دستگاه های پایانه متعدد و لبه های متعدد، سیستم لبه می تواند از عدم تعادل بار رنج ببرد و در صورت عدم وجود معقول بر عملکرد زمانبندی تأثیر بگذارد. از این رو ضرورت استفاده از راهکارهای نوین در فرایندهای زمانبندی منابع در محیط لبه قابل توجه است.

# 4-1) جنبه نوآوری پژوهش

محاسبات لبه / مه، به عنوان یک الگوی محاسباتی توزیع شده، الزاماتی شامل تأخیر پایین و قابلیت پردازش بلادرنگ را برای تعداد روزافزونی از برنامه های اینترنت اشیا برآورده می کند و به پارادایم محاسباتی اصلی در ساختار اینترنت اشیا تبدیل شده است. با این حال، از آنجایی که تعداد زیادی از برنامه های اینترنت اشیا نیاز به اجرا در منابع لبه / مه دارند، ممکن است سرورها بیش از حد بارگذاری شوند. از این رو، ممکن است سرورهای لبه / مه را مختل کند و همچنین بر زمان پاسخ سیستم تأثیر منفی بگذارد[10]. علاوه بر این، بسیاری از برنامه های کاربردی اینترنت اشیا از اجزای وابسته تشکیل شده اند که محدودیت های اضافی را برای اجرای خود متحمل می شوند. علاوه بر این، محیط های محاسباتی لبه / مه و برنامه های اینترنت اشیا ذاتاً پویا و تصادفی هستند. بنابراین، ارائه یک راهکار زمان بندی کارآمد و سازگار برنامه های اینترنت اشیا در محیط های محاسباتی لبه / مه ناهمگن از اهمیت بالایی برخوردار است[2]. با این حال، منابع محاسباتی محدود سرورهای لبه و مه بار اضافی را برای اعمال تکنیک های بهینه اما محاسباتی تحمیل می کنند. برای غلبه بر این چالش ها، در این پژوهش یک رویکرد زمانبندی بهینه مبتنی بر یادگیری عمیق با استفاده از شبکه عصبی LSTM پیشنهاد می شود تا زمان پاسخ را به طور سازگار و کارآمد بهینه کند. به طور کلی، جنبه نوآوری پژوهش به شرح زیر است:

* در این پژوهش از یک رویکرد پیش پردازش و حذف افزونگی اطلاعات مبتنی بر مدل پیرسون به منظور مدیریت منابع با استفاده از پارامترهای کیفیت سرویس ارائه می شود. هنگامی که محیط محاسباتی تغییر می کند (به عنوان مثال، درخواست از برنامه های مختلف اینترنت اشیا، منابع محاسباتی سرور، تعداد سرورها)، این راهکار می تواند به طور تطبیقی خط مشی مربوط به پارامترهای کیفیت سرویس و زمان بندی را با سرعت همگرایی بالا به روز کند.
* در ادامه به منظور زمانبندی وظایف، یک مدل هزینه مبتنی بر شبکه عصبی LSTM در محیط های محاسباتی لبه / مه پیشنهاد می شود تا سطح تعادل بار سرورها را بهبود بخشد و زمان پاسخ برنامه را به حداقل برساند.

# 5-1) اهداف پژوهش

* بهینه‌سازی تعادل بار سیستم در محیط‌های محاسباتی لبه و مه
* کمینه‌سازی زمان پاسخ سیستم در محیط‌های محاسباتی لبه و مه
* بهبود سربار محاسباتی در محیط‌های محاسباتی لبه و مه
* بهبود نرخ همگرایی در محیط‌های محاسباتی لبه و مه

# 6-1) سؤالات پژوهش

* راهکار زمانبندی پیشنهادی با استفاده از چه فرایندی باعث بهینه‌سازی تعادل بار سیستم در محیط‌های محاسباتی لبه و مه می شود؟
* رویکرد پیشنهادی زمانبندی با استفاده از چه راهکاری باعث بهینه‌سازی زمان پاسخ سیستم در محیط‌های محاسباتی لبه و مه می شود؟
* راهکار زمانبندی پیشنهادی با استفاده از چه فرایندی موجب بهینه‌سازی باعث بهبود سربار محاسباتی در محیط‌های محاسباتی لبه و مه می شود؟
* رویکرد پیشنهادی زمانبندی با استفاده از چه راهکاری باعث بهبود نرخ همگرایی در محیط‌های محاسباتی لبه و مه می شود؟

# 7-1) فرضیات پژوهش

* راهکار زمانبندی با به‌کارگیری یک رویکرد مبتنی بر یادگیری عمیق و بهبود فرایند تخصیص منابع باعث بهینه‌سازی تعادل بار سیستم در محیط‌های محاسباتی لبه و مه می شود.
* راهکار زمانبندی پیشنهادی با به‌کارگیری یک رویکرد مبتنی بر یادگیری عمیق و انتخاب بهینه‌منابع باعث بهینه‌سازی زمان پاسخ سیستم در محیط‌های محاسباتی لبه و مه می شود.
* راهکار زمانبندی پیشنهادی با استفاده از یک رویکرد بهینه‌سازی زمانبندی در محیط‌های محاسباتی لبه و مه مبتنی بر یادگیری عمیق موجب بهینه‌سازی باعث بهبود سربار محاسباتی می شود.
* رویکرد پیشنهادی زمانبندی با به‌کارگیری راهکار یادگیری عمیق و کاهش سربار شبکه، موجب بهبود نرخ همگرایی در محیط‌های محاسباتی لبه و مه می شود.

# 8-1) توصیف مفاهیم اصلی پژوهش

* **رایانش ابری سیار:** رایانش ابری سیار به این معنی است که دستگاه های تلفن همراه قادر به دسترسی به ابر خواهند بود. یک زیرساخت ایده‌آل رایانش ابری سیار می‌تواند مصرف انرژی، ذخیره‌سازی، ارتباطات، امنیت و قابلیت‌های محاسباتی قابل مقایسه با دستگاه های تلفن همراه را فراهم کند[11].
* رایانش ابری موردی سیار: یک زیرساخت ابر متمرکز باید قبل از اینکه محاسبات ابری سیار به‌عنوان مدل تحویل برنامه به کار گرفته شود[11].
* **محاسبات لبه:** خدمات (به‌عنوان مثال، پردازش و ذخیره‌سازی) را می‌توان از هسته به لبه شبکه منتقل کرد. محاسبات لبه، محاسبات را در لبه شبکه انجام می دهد، نه در دستگاه های پایانی (یا منابع داده). در نزدیکی دستگاه های تلفن همراه خود دستگاه های تلفن همراه می‌توانند از این فناوری برای دسترسی به داده ها استفاده‌کنند، حتی اگر به اینترنت متصل نباشند[12].
* **محاسبات مه:** محاسبات ابری به محاسبات، ذخیره‌سازی و قابلیت‌های شبکه اجازه می دهد تا به دستگاه های اینترنت اشیا (یا چیزهایی) کاهش یابند تا استقلال را افزایش دهند. در محاسبات مه، محاسبات انجام‌می شود، داده ها ذخیره می شوند و شبکه‌ها در لبه شبکه‌های ارتباطی ایجاد می شوند[12].
* **رایانش ابری:** توسعه محاسبات ابری برای رفع نواقص محاسبات لبه موبایل و رایانش ابری سیار در نظر گرفته شده بود. با منابع مناسب و اتصالات اینترنتی قوی، یک ابر می‌تواند با کمک یک کامپیوتر یا دسته‌ای از رایانه‌ها، به دستگاه های تلفن همراه مجاور خدمات ارائه دهد. پوشش ابرها محدود است. در رایانش ابری، منابع محاسباتی در شبکه‌ای از رایانه‌های مورد تقاضا به اشتراک گذاشته می شوند و در همه‌جا در دسترس هستند. از آنجایی که دستگاه های اینترنت اشیا حجم عظیمی از داده ها را تولید می‌کنند، محاسبات ابری پردازش آن داده ها را آسان‌تر و مؤثرتر می‌کند و خدمات پرداخت به‌صورت درخواستی را ارائه می دهد[12].

# 9-1) ساختار پایان‌نامه

سازمان دهی پایان نامه پیش رو در پنج فصل ارائه شده است. در فصل اول بیان مسئله، حیطه مطالعه با عنوان زمانبندی در محاسبات لبه و مه، چالش موجود، اهمیت و ضرورت تحقیق، جنبه نوآوری پژوهش به همراه اهداف و فرضیات پژوهش بیان شدند. در فصل دوم ادبیات و پیشینه ای از ساختار شبکه های مه و لبه، چالش ها و مسئله زمانبندی در آن مورد بررسی قرار گرفته و به ارزیابی مزایا و معایب هر یک از راهکارهای ارائه شده پرداخته خواهد شد. در فصل سوم، رویکرد پیشنهادی زمانبندی در شبکه های مه و لبه ارائه شده و ساختار کلی آن مورد ارزیابی و بررسی قرار می گیرد. در ادامه و در فصل چهارم روش پیشنهادی از نظر پارامترهای مختلف مورد ارزیابی و بررسی قرار گرفته و الگوریتم و فلوچارت مدل پیشنهادی بیان می شود. در فصل پنجم ضمن توصیف نتایج به دست آمده، پیشنهادات کلی در این زمینه توصیف خواهد شد.

2) فصل دوم

**پیشینه پژوهش**

**اهداف فصل:**

* توصیف ادبیات پژوهش
* توصیف راهکارهای موجود و پیشینه پژوهش
* مقایسه روش‌های موجود

# 1-2) مقدمه

با توجه به گستردگی شبکه های نرم افزار محور، برای اینکه بتوان در سطح وسیعی، از شبکه بهره برد، به راهکارهای مفیدی برای ایجاد تعادل بار در این شبکه ها نیاز است. لذا گسترش این سیستم ها و افزایش استفاده از آن ها، مسئله تعادل بار را به یک چالش مهم در جهت دستیابی به کارایی بالا در این سیستم ها تبدیل کرده است. این فصل جهت آشنایی بیشتر با روش های ایجاد تعادل بار در شبکه های نرم افزار محور تنظیم شده است. در ابتدا روش های موجود مورد مطالعه و بررسی قرار می گیرند. سپس، مزایا و معایب نمونه هایی از آن ها بررسی شده و با یکدیگر مقایسه می شوند.

# 2-2) ادبیات پژوهش

در این بخش از پژوهش به بررسی و ارزیابی ساختارهای مربوط به محاسبات مه و لبه و فرایند زمانبندی وظایف جهت ارتقا کیفیت سرویس پرداخته خواهد شد. برای برنامه های حساس به تأخیر، پاسخ های دیرهنگام ممکن است منجر به خطا در سیستم یا حتی خرابی سیستم شود. به عنوان مثال، در یک برنامه کاربردی سلامت هوشمند که وضعیت بیمار را کنترل می کند، پاسخ های دیرهنگام ممکن است زندگی انسان را به خطر بیندازد. وقتی حجم زیادی از داده های تولید شده توسط دستگاه های اینترنت اشیا را در نظر بگیریم، این امر حتی مهم تر می شود. به عنوان مثال، در برنامه های شهر هوشمند، دستگاه های اینرنت اشیا حدود یک میلیون رکورد در ثانیه تولید می کنند. با توجه به تراکم بالای سنسورها در تعدادی از کاربردها و دقت کم اندازه گیری، امکان تکرار و خطاپذیری داده ها در محیط های اینرنت اشیا قابل توجه است. بنابراین، این داده های تکراری و مستعد خطا ممکن است اندازه داده های غیر ضروری را افزایش دهد. انتقال حجم عظیمی از داده ها و درخواست ها به ابر ممکن است منجر به استفاده کم از منابع شبکه، تأخیر زیاد در انتقال، هزینه های مالی و سربار پردازش و ازدحام شبکه شود.

# 1-2-2) فرایندهای زمانبندی در محاسبات مه

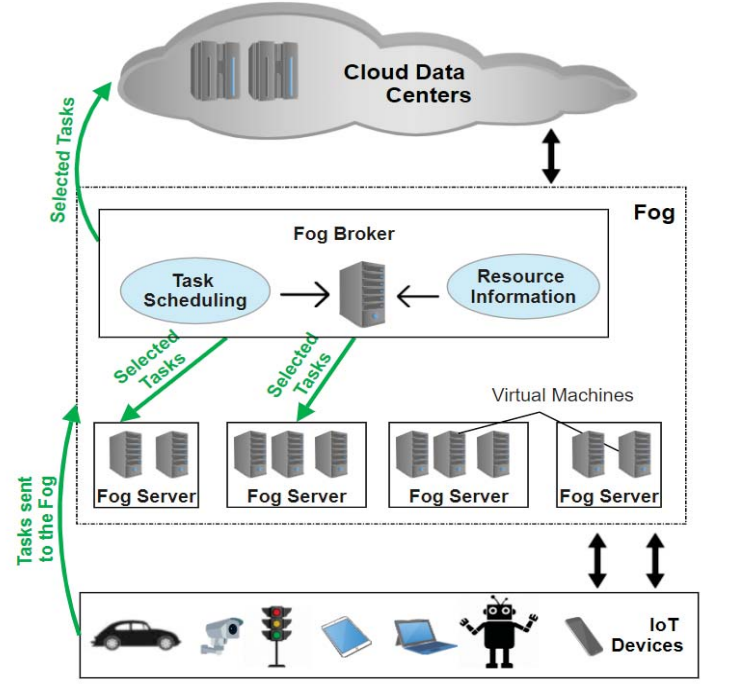
برای برنامه های حساس به تأخیر، پاسخ های دیرهنگام ممکن است منجر به خطا در سیستم یا حتی خرابی سیستم شود. به عنوان مثال، در یک برنامه کاربردی سلامت هوشمند که وضعیت بیمار را کنترل می کند، پاسخ های دیرهنگام ممکن است زندگی انسان را به خطر بیندازد. وقتی حجم زیادی از داده های تولید شده توسط دستگاه های اینترنت اشیا را در نظر بگیریم، این امر حتی مهم تر می شود. به عنوان مثال، در برنامه های شهر هوشمند، دستگاه های IoT حدود یک میلیون رکورد در ثانیه تولید می کنند. با توجه به تراکم بالای سنسورها در تعدادی از کاربردها و دقت کم اندازه گیری، امکان تکرار و خطاپذیری داده ها در محیط های اینترنت اشیا قابل توجه است. بنابراین، این داده های تکراری و مستعد خطا ممکن است اندازه داده های غیر ضروری را افزایش دهد. انتقال حجم عظیمی از داده ها و درخواست ها به ابر ممکن است منجر به استفاده کم از منابع شبکه، تأخیر زیاد در انتقال، هزینه های مالی و سربار پردازش و ازدحام شبکه شود[13].

به گفته محققان، محاسبات مه به عنوان یک راه حل امیدوارکننده، یک پلتفرم بسیار مجازی است که ظرفیت های پردازش، ذخیره سازی و شبکه را بین دستگاه های اینترنت اشیا و منابع رایانش ابری فراهم می کند و به طور کلی، اما نه منحصراً، مه در لبه شبکه قرار دارد. یک معماری رایج محاسبات مه شامل سه لایه اصلی است: دستگاه، مه و لایه ابر. محاسبات مه یک لایه میانی بین لایه های دستگاه و ابر فراهم می کند که شامل تعداد زیادی گره مه است که قابلیت پردازش، شبکه سازی و غیره را دارند. از آنجایی که قرار است گره های مه در نزدیکی لایه دستگاه قرار گیرند، درخواست های دارای الزامات تأخیر بالا را می توان در لایه مه پردازش کرد و سایر درخواست ها را می توان به ابر ارسال کرد. علاوه بر این، از آنجایی که قابلیت های پردازش محلی را فراهم می کند، می تواند استفاده از پهنای باند و هزینه مالی را کاهش دهد. لایه مه، درخواست ها و داده های دریافتی را فیلتر، تجزیه و تحلیل و پیش پردازش می کند. اگر لایه مه منابع کافی در دسترس داشته باشد و بتواند وظیفه ای را اجرا کند، کار در لایه مه برنامه ریزی می شود. در غیر این صورت، اگر منابع کافی برای اجرای کار وجود نداشته باشد، به لایه ابر ارسال می شود. اگر هیچ یک از آنها (یعنی لبه و مه) نتوانند در زمان مشخص شده پاسخ دهند، کار رد می شود. بنابراین، یافتن مکان بهینه خدمات و منابع در معماری اینترنت اشیا، مولفه هایی برای دستیابی به بهبود هزینه و کارایی منابع، کیفیت سرویس بالاتر و سطح بالاتری از حریم خصوصی امنیتی مهم است. علاوه بر این، لایه مه ممکن است به طور همگن یا ناهمگن سازماندهی شود[14].

در ساختار همگن، منابع مه دارای قابلیت های پردازش، ذخیره سازی و پهنای باند یکسانی هستند، اما در حالت ناهمگن، منابع دارای قابلیت های متفاوتی هستند. در مورد دستگاه های مه ناهمگن، زمان بندی وظایف از پیچیدگی بیشتری برخوردار است، زیرا منابع دارای قابلیت های متفاوتی از نظر سرعت پردازش، جابه جایی و غیره هستند که منجر به ترکیب های مختلفی (وظیفه و منبع) می شود. جدا از زمان پاسخگویی و ضرب الاجل برای هر کار، پارامترهای دیگری مانند بهره وری منابع، انرژی و هزینه مالی نیز ممکن است لازم باشد توسط الگوریتم های زمان بندی کار در نظر گرفته شوند. در نظر گرفتن چندین پارامتر در کنار ناهمگونی وظایف و منابع، پیچیدگی بیشتری را بر فرآیند تصمیم گیری تحمیل می کند. با افزایش تعداد کارها و منابع، راه حل های ممکن به صورت تصاعدی رشد می کنند و هیچ الگوریتم چند جمله ای قطعی برای حل این مسئله وجود ندارد و به عنوان یک مسئله NP-Hard در نظر گرفته می شود[15].

از طرفی دیگر، برنامه های کاربردی اینترنت اشیا (IoT) معمولاً از پلتفرم رایانش ابری به دلیل قدرت محاسباتی فراوان و قابلیت های ذخیره سازی داده ها استفاده می کنند. با رشد سریع اینترنت اشیا و برنامه های کاربردی بلادرنگ، محاسبات ابری برای برنامه های کاربردی اینترنت اشیا حساس به زمان به دلیل تأخیر ارتباطی زیاد بین مراکز داده ابری و دستگاه های اینترنت اشیا مناسب نیست. هدف محاسبات مه کاهش تأخیر شبکه با ارائه منابع مورد نیاز در لبه شبکه نزدیک به دستگاه های پایانی است، بنابراین به کاهش تأخیر، بهبود کیفیت خدمات و تحرک پشتیبانی کمک می کند. با این حال، منابع مه محدود هستند و نمی توانند حجم عظیمی از داده های تولید شده توسط دستگاه های IoT را به صورت محلی پردازش کنند. بنابراین، انتخاب مؤثر پردازش وظایف بین سرور ابری و سرور مه برای برآوردن نیازهای روزافزون و سفارشی برنامه های اینترنت اشیا مهم است. در مطالعات موجود، چندین الگوریتم برای تقسیم وظایف بین لبه و مه بر اساس پارامترهایی مانند زمان اجرا و مهلت وظایف، طول کار، امنیت و مهلت وظایف پیشنهاد شده است. با این حال، این کارها به اندازه کافی نیازهای منابع وظایف را با در نظر گرفتن منابع محدود در دسترس بودن مه برآورده نمی کنند. علاوه بر این، منابع مورد نیاز وظایف را نمی توان به طور دقیق در زمان ارسال کار تعیین کرد[16].

بنابراین، اختصاص دادن یک کار به یک گره پردازشی مناسب در لایه مه در یک محیط ابر-مه ناهمگن بسیار مهم است. یک گره لبه تلاش می کند تا یک وظیفه را به صورت محلی اجرا کند و در صورت شکست آن را به سایر گره های لبه یا ابر ارسال کند. با این حال، در یک محیط مه (مثلاً گره لبه) با ماشین های مجازی ناهمگن، معیار انتخاب ماشین مجازی مناسب (برای وظایف) یک نیاز مهم برای بهبود کارایی کلی اجرای کار است. علاوه بر این، یک کارگزار مه در لایه مه برای مدیریت منابع مه و برنامه ریزی وظایف در گره های مه، بسته به پارامترهایی مانند تقاضا و در دسترس بودن منابع و تحمل تأخیر استفاده می شود. برای این منظو، یک الگوریتم زمان بندی کار بلادرنگ در کارگزار مه با در نظر گرفتن مهلت و منابع مورد نیاز وظایف علاوه بر در دسترس بودن منابع و حجم کار گره های مه، نیاز است. به طور خاص، الگوریتم زمانبندی باید یک چارچوب ارتباطی و اجرای فرایند کارآمد برای محاسبات مه ایجاد کند تا از وظایف سفارشی شده مبتنی بر اینترنت اشیا را پشتیبانی کند که در آن ملاحظات مهم شامل محدودیت های زمانی وظایف، منابع موجود در لایه مه، و تأخیر بین لایه های لبه و مه است. در شکل زیر ساختار یک فرایند زمانبندی در محاسبات لبه و مه ارائه شده است.



**شکل 2- 1) چارچوب معماری محاسبات مه-لبه با زمان‌بندی وظایف[1]**

به طور کلی، انتقال جریان کاری به محیط محاسبات لبه و مه، به کار بردن سرویس های ابری مختلف را برای تسهیل اجرای جریان کار امکان پذیر می کند. مثال های عمومی جریان کار شامل بانکداری آنلاین ، پردازش بیمه ، خانه های هوشمند و بسیاری از کارهای دیگر می باشد. هدف اصلی سیستم مدیریت جریان کار پشتیبانی از تعریف، اجرا، ثبت و کنترل فرایندهای کسب و کار است. در زمان اجرا یک موتور اجرای جریان کار، اجرای جریان کار را با به کار بردن میان افزارها مدیریت می کند. اجزاء اصلی در یک موتور اجرای جریان کار، زمانبندی جریان کار، اجرای جریان کار، انتقال داده ها و مدیریت خطاها هست. زمانبند جریان کار منابع را کشف کرده و وظایف را به منابع مناسب اختصاص می دهد. انتقال داده ها برای ارتباط منابع داده و مدیریت خطا برای مدیریت خطاها در طی اجرا به کار می رود. زمانبندی جریان کار نقش حیاتی در مدیریت جریان کار ایفا می کند. زمانبندی صحیح می تواند اثر قابل توجهی روی کارایی سیستم داشته باشد. برای زمانبندی مناسب در جریان کار، الگوریتم های زمانبندی مختلفی باید مورد بحث قرار گیرد. به طور کلی برنامه های جریان کاری به صورت یک دنباله اجرا می شوند زیرا به صورت یک مجموعه از وظایف مختلف برای رسیدن به یک نتیجه ی خاص در نظر گرفته شده اند. وظایف بر اساس وابستگی به داده ها اجرا می شوند، آن را به صورت یک گراف در نظر بگیرید که ابتدا والدها و درنهایت فرزندان باید اجرا شوند[1].

زمانبندی در محیط لبه و مه به معنی انتخاب بهترین منبع برای یک وظیفه است یا اختصاص ماشین های مجازی به وظایف به گونه ای که زمان اجرای کار به حداقل برسد. به طور کلی در الگوریتم های زمانبندی لیستی از وظایف وجود دارد که به ترتیب اولویت برای همه ی وظایف ساخته شده است . وظایف بر اساس اولویت انتخاب می شوند و به پردازنده ها تخصیص داده شده که تابع هدف آنها از پیش تعیین شده است. روش های مختلفی برای مشکل زمانبندی در سیستم های لبه و مه وجود دارد که در ادامه تعدادی از این رویکردها ارائه می شود[1]:

* **الگوریتم های زمانبدی ایستا و پویا:** دو نوع الگوریتم زمانبندی با نامهای ایستا و پویا وجود دارد. در زمان بندهای ایستا اطلاعات لازم در مورد همه منابع در لحظه ی زمانبندی موجود است به همین دلیل زمان اجرا را می توان تخمین زد، در مقابل آن در زمان بندهای پویا ایده اصلی تخصیص وظایف در زمان اجرا می باشد.
* **الگوریتم های زمانبندی سراسری و محلی** : در زمانبندی محلی در مورد تخصیص فرایندها به یک پردازنده ولی در زمانبندی سراسری تخصیص فرایندها به چندین پردازنده تصمیم گیری می شود.
* **الگوریتم های توزیع شده و متمرکز**: در زمانبندی توزیع شده، مدیر مرکزیوجود ندارد که مسئول اجرای باشد و بلکه هر سیستم بر اساس الگوریتم توزیع شده به فعالیت می پردازد، ولی در زمان بندیهای متمرکز یک کنترل مرکزی برای تخصیص تمامی وظایف وجود دارد.
* **الگوریتم های زمانبندی تقریبی و اکتشافی** : الگوریتم های تقریبی راه حل هایی شبه بهینه همراه با ضریب برای میزان تقویت جواب واقعی ارائه می دهند ولی الگوریتم های اکتشافی با فرض های واقعی در مورد سیستم و جست و جوی فضای مسئله ، سعی در پیدا کردن جوابی نزدیک به جواب بهینه را دارد.
* **الگوریتم های زمانبندی همکار و غیر همکار:** الگوریتم های زمانبندی همکار با اجزای توزیع شده سروکار دارند و در ارتباط هستند و در زمان بندیهای غیر همکار هر پردازنده مستقل از سایر پردازنده ها است
* **الگوریتم های زمانبندی بهینه و نیمه بهینه:** روش بهینه سعی دارد هدف را به بهترین شکل برآورده سازد در صورتی که روش های نیمه بهینه تنها به یک هدف نمی پردازد بلکه سعی بر بهبود نسبی چندین هدف را به صورت همزمان دارند.

# 2-2-2) چالش‌های محاسبات مه و لبه

* اگرچه محاسبات ابری پتانسیل ارائه مزایای قانع کننده را در بسیاری از موارد استفاده دارد، اما این فناوری به دور از خطا نیست. فراتر از مشکلات سنتی محدودیت های شبکه، چندین ملاحظات کلیدی وجود دارد که می تواند بر پذیرش محاسبات ابری تأثیر بگذارد: استقرار یک زیرساخت در مه و لبه می تواند مؤثر باشد، اما دامنه و هدف استقرار لبه باید به وضوح تعریف شود. یک استقرار محاسبات ابری گسترده، با استفاده از منابع محدود و خدمات کم، هدف خاصی را در مقیاس از پیش تعیین شده انجام می دهد. قابلیت اتصال محاسبات ابری بر محدودیت های معمولی شبکه غلبه می کند. طراحی یک استقرار لبه ای که اتصال ضعیف یا نامنظم را در خود جای دهد و در نظر گرفتن آنچه در لبه هنگام از بین رفتن اتصال رخ می دهد، بسیار مهم است. استقلال، هوش مصنوعی و برنامه ریزی موفق شکست در پی مشکلات اتصال برای محاسبات ابری موفق ضروری است. طراحی یک استقرار محاسبات ابری ای که بر مدیریت صحیح دستگاه، مانند اجرای پیکربندی مبتنی بر سیاست، و همچنین امنیت در منابع محاسباتی و ذخیره سازی از جمله عواملی مانند وصله نرم افزار و به روزرسانی ها تأکید می کند، حیاتی است. مشکل همیشگی افزایش داده های امروزی این است که بسیاری از این داده ها غیر ضروری هستند. برخی از چالش های استفاده از محاسبات ابری در ادامه توضیح داده شده است[17].
* **زیرساخت نرم افزار:** تحول به سمت مدل های مجازی و بومی ابری اساساً زیرساخت های لبه را به مدل استقرار به عنوان یک سرویس با استفاده از سازمان دهنده های ابر استاندارد صنعتی تبدیل کرده است. توابع شبکه اختصاصی در حال حاضر نیاز به تکامل به سمت معماری مبتنی بر میکرو سرویس برای مدل های استقرار سرویس گرا دارند[17].
* **قابلیت مدیریت یکپارچه در سراسر لبه:** محاسبات ابری را می توان بر اساس نوع ترافیک، برنامه های کاربردی در حال سرویس، اتصال دستگاه و نقطه حضور به چند بخش تقسیم کرد، تعداد مناطق محاسباتی لبه ای که در سراسر مناطق جغرافیایی گسترده می شوند، افزایش می یابد. هر یک از مناطق محاسباتی لبه ای که به ناحیه خاصی از منطقه جغرافیایی یا مجموعه ای از پهنای باند شبکه می پردازد، به مکانیزم قابل همکاری با سایر مناطق محاسبات ابری ای نیاز دارد تا اتصال یکپارچه را فراهم کند. این نیاز به هماهنگی یکپارچه و مدیریت چرخه زندگی در این خوشه های متعدد و مناطق ابری را ایجاد می کند[17].
* **ابر عمومی و خصوصی**: استفاده از فناوری های بومی ابری در سناریوهای چند لبه اساساً ارائه دهنده خدمات را مجبور می کند تا خوشه های ابر خصوصی را در این لبه ها فعال کند. با این حال، ارائه دهندگان خدمات ابر مقیاس بزرگ مانند مایکروسافت آژور یا خدمات وب آمازون و غیره، فرصت اقتصاد ابر مقیاس را با استفاده از ساختارهای ابر عمومی فراهم می کنند. اکنون می توان زیرساخت را به شیوه ای هوشمندانه و مقرون به صرفه مقیاس بندی کرد و در عین حال از رابط برنامه ریزی پذیر برنامه (API) یکپارچه در سراسر زیرساخت ارائه شده توسط ارائه دهنده ابر عمومی استفاده کرد[17].
* **امنیت و حریم خصوصی:** به دلیل ماهیت متنوع انواع مختلف استقرار لبه ها، نیازهای یک لبه ایمن به مجموعه ای از رویکردهای چند وجهی تبدیل شده است که باید در هر استقرار لبه سفارشی شود. توانایی ارائه مجوز احراز هویت و حسابداری در لبه توزیع شده نیازمند سیاست های چند سطحی است تا اطمینان حاصل شود که هر کاربر نهایی در نظر گرفته شده است. حفظ حریم خصوصی کاربر نهایی یا دستگاه نهایی یکی دیگر از جنبه های حیاتی است که باید هنگام عبور ترافیک از لبه حفظ شود. معماری امنیتی صفر اعتماد یکی از جدیدترین الگوهایی است که بر این باور است که هیچ جنبه ای از ارتباطات داده ایمن نیست زیرا هیچ شخصیت قابل اعتمادی وجود ندارد. اجرای این موارد با الزامات تأخیر کم در زمان واقعی همچنان یک چالش بزرگ است[17].
* **انتزاع و استفاده از سخت افزار:** خلاصه کردن تمام ویژگی های سخت افزار، شتاب دهنده ها و هر پیشرفت دیگری که برای توابع شبکه مجازی یا میکروسرویس های بومی ابری موجود است، ارزش پیشنهادی سخت افزار COTS است. با ظهور مدل های سخت افزاری مختلف به عنوان سرویس مانند واحد پردازشگر گرافیکی به عنوان یک سرویس یا واحد پردازشگر زیرساخت یا به طور کلی واحد پردازشگر به عنوان یک سرویس، باید هوشمندی اضافی برای عملکردهای شبکه حساس به تأخیر برای استفاده کامل از ویژگی های سخت افزاری مختلف با قابلیت مقیاس پذیری در سراسر استقرار لبه داشته باشد[17].
* **ارزش داده ها:** الگوی محاسبات ابری توزیع شده و تجمیع شده، حجم عظیمی از داده را در زیرساخت لبه انتها به انتها به ارمغان می آورد. با این حال، چالش این است که ارزش داده ها با افزایش زمان تأخیر بیشتر از نقطه مبدا کاهش می یابد. بنابراین، مکانیزم های پردازش و تحلیل داده کارآمد و با تأخیر کم در لبه های توزیع شده که به کاربران نهایی نزدیک تر هستند، مورد نیاز است.
* **مدل های هوش مصنوعی و یادگیری ماشین برای لبه:** برای به دست آوردن ارزش داده های تولید شده در لبه های مختلف، مقادیر عظیمی از نقاط داده مرتبط با یک کاربر نهایی یا یک دستگاه واحد نیاز به پردازش و تجزیه و تحلیل در یک بازه زمانی ثابت دارند. این امر مستلزم مدل های یادگیری ماشین و هوش مصنوعی سفارشی است که باید برای هر یک از نوع لبه ها و سیستم مدیریتی تنظیم شوند که بتوانند مدل مناسب را برای موارد استفاده ضروری اعمال کنند. دامنه وسیعی برای نوآوری و توسعه در این فضا برای موارد استفاده مختلف در لبه ها وجود دارد[18].
* **مدیریت چرخه حیاط:** توابع شبکه مبتنی بر نرم افزار در سراسر زیرساخت لبه مستقر می شوند و اپراتور را ملزم می سازد تا با بسته بندی، نصب و استقرار عملکردهای شبکه مقابله کند[19].
* **مدیریت سیاست:** خط مشی های در لبه مجموعه ای از قوانین و محدودیت هایی هستند که بر استقرار سرویس های لبه توسط افراد مختلف مانند مدیران، ارائه دهندگان خدمات، توسعه دهندگان برنامه ها، صاحبان سرویس، پرسنل عملیات و غیره کنترل می کنند. این خط مشی ها به سازمان دهندگان مبتنی بر ابر کمک می کنند تا محدودیت های هر یک را درک کنند نوع برنامه مانند محدودیت های سخت افزاری، تحمل تأخیر، اولویت برنامه، مدل های اجرا تا تکمیل، الزامات امنیتی، الزامات مقیاس و غیره. سیاست ها بر اساس نوع لبه بسیار متفاوت است. در نهایت یک مدیر خط مشی متمرکز مورد نیاز است که بتواند با مدیران و مجریان خط مشی فردی در هر یک از انواع لبه تعامل داشته باشد[20].
* **اتوماسیون شبکه:** مدیریت و حفظ ظرفیت محاسباتی توزیع شده و ماهیت متفاوت الزامات عملکرد شبکه در محیط های لبه نیازمند 100 عملیات در هر لحظه است. این در معماری مبتنی بر شبکه های نسل پنجم ارتباطی که در آن اطلاعات در شبکه به طور گسترده توزیع شده است، بسیار پیچیده تر می شود. اتوماسیون عملیات شبکه یک عامل متمایز کننده بزرگ در مالکیت، نگهداری و راه اندازی شبکه در مقیاس است[17].

# 3-2-2) بارگذاری وظایف در محاسبات لبه و مه

تاکنون محاسبات لبه و مه به عنوان یک راه حل کارآمد برای بهبود فرایندهای محاسباتی در نظر گرفته شده اند. زیرا سرورهای ابری قدرت محاسباتی و منابع ذخیره سازی بسیار زیادی دارند، بنابراین بارگذاری وظایف محاسباتی فشرده به سرورهای ابری بار محاسبات ابری را کاهش می دهد. و انرژی حیاتی را ذخیره می کند. با این حال، سرورهای ابری به دلیل موقعیت جغرافیایی خود، به دلیل تأخیر طولانی انتشار بین ماشین های پردازشی و سرورهای ابری، مقدار خوبی از تأخیر و لرزش را معرفی می کنند. همچنین در صورتی که ماشین های پردازشی تمام وظایف خود را به همراه داده های تولید شده با ازدحام شبکه در فضای ابری تخلیه کنند، زمان تکمیل کار را می توان بیشتر کرد. محاسبات ابری یک پارادایم محاسباتی جدید است که اخیراً در عمل برای رفع این مشکلات معرفی شده است. اصل اساسی محاسبات ابری این است که توان محاسباتی ابری را به لبه شبکه نزدیک به ماشین های پردازشی برساند. برنامه های در حال اجرا بر روی ماشین های پردازشی اکنون می توانند وظایف محاسباتی فشرده را به سرورهای محاسبات ابری مجاور با سربار کم تخلیه کنند، که می تواند به طور قابل توجهی مشکل ازدحام شبکه را کاهش دهد و زمان پاسخگویی برنامه های در حال اجرا را بهبود بخشد

سرورهای محاسبات ابری در لبه شبکه در ایستگاه های پایه سلولی یا سلول های کوچک یا هر دو مستقر هستند. طیف گسترده ای از برنامه های کاربردی تلفن همراه امروزه تمایل دارند از خدمات محاسبات ابری برای ارائه تجربه با کیفیت بهتر به کاربران استفاده کنند. برنامه های نوظهور مانند تحویل و ذخیره محتوای توزیع شده، بهبود عملکرد وب، آگاهی از زمینه، بهینه سازی محتوا، و خدمات مختلف شهر هوشمند در حوزه های مراقبت های بهداشتی، حمل و نقل، انرژی، تولید و اتوماسیون صنعتی به شدت به زیرساخت محاسبات ابری برای تخلیه وظایف نیاز دارد.

هدف اصلی از بارگذاری وظیفه به حداقل رساندن زمان اجرای وظیفه برنامه های کاربردی در حال اجرا در سیستم پردازشی صرفه جویی در مصرف انرژی سیستم پردازشی است. مشکل بارگذاری کارآمد کار چندگانه است زیرا سرورهای محاسبات ابری به اندازه سرورهای ابری کارآمد نیستند، بنابراین تخلیه وظایف از وظایف پردازشی به سرورهای محاسبات ابری باید به گونه ای انجام شود که منابع سرورهای محاسبات ابری به طور عاقلانه استفاده شوند. سرورهای محاسبات ابری که در ایستگاه های پایه مختلف میزبانی می شوند ممکن است به مجموعه خدمات یکسانی مجهز نباشند و بار محاسباتی در سرورهای محاسبات ابری نیز ممکن است به صورت پویا با زمان تغییر کند. تخصیص بهینه منابع برای وظایف بارگذاری شده، توزیع بار منصفانه و پایدار بین سرورهای محاسبات ابری، و هماهنگی یکپارچه بین ابر و لبه تنها یکی از جنبه های مشکل برای استفاده مؤثر از خدمات محاسبات ابری است. چالش دیگر در درک ابعاد وظایف تعیین شده برای تخلیه باقی می ماند. بسته به پارامترهای کیفیت سرویس مورد نیاز برای برنامه های در حال اجرا، وظایف نیز می توانند از نظر اولویت ها، پیشرفت، سربار محاسباتی و سایر وابستگی های مرتبط متفاوت باشند. انتخاب، زمان بندی و قرار دادن وظایف شامل مجموعه دیگری از چالش های مرتبط با مشکل تخلیه کار است. بنابراین مشکل تخلیه کار بهینه به انتخاب وظیفه مناسب برای بارگذاری به سرور محاسبات ابری مناسب در زمان مناسب خلاصه می شود تا عملکرد بهینه یک برنامه در حال اجرا را برآورده کند و در عین حال از منابع محاسبات ابری به طور مؤثر استفاده کند. این به وضوح یک مسئله بهینه سازی چند هدفه است و به عنوان مسئله NP-hard شناخته می شود.

بارگذاری وظیفه در زمینه بحث این پژوهش می تواند به عنوان بارگذاری یک ماژول کامل از یک برنامه کاربردی که شامل محاسبات، داده های مورد نیاز و سایر کتابخانه های وابسته به یک سرور راه دور و به دست آوردن نتیجه محاسبه از سرور راه دور است، تعریف شود. در حین بحث در مورد بارگذاری کار، تعریف زمان بندی نیز مهم است، زیرا بخشی ضروری از بارگذاری کار است. زمانبندی روشی است که مجموعه ای از وظایف (یا وظایف فرعی) را بر روی مجموعه ای از منابع محاسباتی برای بهینه سازی یک هدف اجرا می کند. یکی از اهداف اولیه برنامه ریزی، به حداکثر رساندن استفاده از منابع و در عین حال به حداقل رساندن کل زمان اجرای کار است. یک سرور محاسبات ابری کار را برای اجرا به صورت محلی زمان بندی می کند یا بسته به مدل های مختلف سیستم، کار را به ابر یا دیگر سرور محاسبات ابری ارسال می کند. مفهوم مشابهی مانند بارگذاری وظیفه، بارگذاری محاسباتی است که به معنای ارسال تنها بخش محاسباتی یک ماژول به یک سرور راه دور و دریافت نتیجه محاسبات از سرور راه دور است.

# 3-2) پیشینه پژوهش

در این بخش به بررسی و ارزیابی پیشینه ای از پژوهش های انجام شده در زمینه زمانبندی در شبکه های مه و لبه توصیف شده و مزایا و معایب هر یک از رویکردها بیان خواهد شد.

* به طور کلی یک محیط محاسباتی مه به معنی استقرار برنامه های کاربردی به منظور نزدیک کردن خدمات محاسباتی، ذخیره سازی و شبکه به کاربران برای رفع نیازهای تأخیر و زمان پاسخ است. گروه تحقیقاتی هالوا و همکارانش[1] رویکردی را به منظور بهبود کارایی کلی اجرای وظایف برنامه های کاربردی با انتخاب مناسب وظایف بلادرنگ سفارشی شده برای اجرا در لایه مه ارائه کردند. به طور خاص، آنها یک الگوریتم جای گذاری سرویس کار مبتنی بر منطق فازی را برای تقسیم وظایف بین لایه های مه و ابر در یک چارچوب محاسباتی ابر-مه پیشنهاد کردند. الگوریتم پیشنهادی، واحدهای پردازش مناسب را برای اجرای وظایف ارسالی در لایه مه با منابع ناهمگن و با بهره برداری از الزامات وظایف (مانند محاسبات، ذخیره سازی، پهنای باند) و محدودیت های آن ها (مانند مهلت زمانی، اندازه داده) انتخاب می کند. آزمایش های شبیه سازی کارایی الگوریتم پیشنهادی و عملکرد برتر آن در مقایسه با سایر الگوریتم های موجود از نظر نسبت موفقیت وظایف، زمان انجام، میانگین زمان چرخش و میزان تأخیر را نشان می دهند.
* پارادایم محاسبات مه به دلیل تمرکز بر انتقال فرآیند محاسباتی به لبه شبکه در نزدیکی کاربر نهایی، علاقه تحقیقاتی فزاینده ای را ایجاد کرده است. گره های مه، اکثر دستگاه های فیزیکی در یک محیط مه، از نظر جغرافیایی پراکنده و ناهمگن با منابع محدود هستند. محاسبات مه انرژی زیادی مصرف می کند زیرا تعداد زیادی گره مه با محدودیت انرژی در محیط مه وجود دارد. بهینه سازی کیفیت خدمات برای محیط های یکپارچه اینترنت اشیا و محاسبات مه برای ارائه خدمات مقرون به صرفه و کم مصرف بسیار مهم است. الشماری و همکارانش[7] تحلیلی از آخرین تکنیک های زمان بندی منابع برای محیط های یکپارچه اینترنت اشیا و محاسبات مه ارائه کردند. علاوه بر این، طبقه بندی تکنیک های زمان بندی منابع برای محیط های محاسباتی اینترنت اشیا و مه یکپارچه برای درک وضعیت فعلی آنها و شناسایی شکاف های تحقیقاتی موجود پیشنهاد شده است. علاوه بر این، استفاده از یادگیری فدرال برای بهینه سازی کیفیت سرویس را مورد بحث قرار می دهد. در نهایت، جهت گیری های آتی را برای تحقیق در این زمینه پیشنهاد می کند.
* محاسبات مه از منابع محل کاربر برای ارائه خدمات بهتر نسبت به رایانش ابری سنتی استفاده می کند. با توجه به ناهمگونی دستگاه های مه، زمان بندی یک چالش است. غفاری و همکارانش[10] نسخه جدیدی از بهینه سازی خرگوش های مصنوعی به نام بهینه سازی خرگوش های مصنوعی آشوب زده مبتنی بر غیرخطی را پیشنهاد می کند و از بهینه سازی خرگوش برای زمان بندی کار در محیط محاسبات مه استفاده می کند. روش پیشنهادی، فرایند زمانبندی را با استفاده از پارامترهای کنترل آشفته و غیرخطی بهینه می کند. در روش پیشنهادی، از نقشه های آشوبی برای بهبود رفتار اکتشافی بهینه سازی خرگوش استفاده می شود. رفتارهای اکتشافی و بهره برداری بهینه سازی خرگوش نیز با استفاده از یک پارامتر کنترل غیرخطی تنظیم می شود. سه هدف در نظر گرفته شده است: زمان خدمات، هزینه و مصرف انرژی. با اولویت بندی وظایف بر اساس ضرب الاجل، عملکرد را بهبود می بخشد. یک سناریوی گسترده الگوریتم های بهینه سازی خرگوش و پیشنهادی را با الگوریتم های دیگر مقایسه می کند. بر اساس نتایج مقایسه، الگوریتم های پیشنهادی بهترین نتایج را از نظر زمان ساخت، زمان سرویس، هزینه کل، مصرف انرژی، نرخ انتشار دی اکسید کربن و درصد رضایت مهلت به دست آوردند.
* محاسبات لبه / مه، به عنوان یک الگوی محاسباتی توزیع شده، الزامات کم تأخیر تعداد روزافزون برنامه های اینترنت اشیا را برآورده می کند و به پارادایم محاسباتی اصلی پشت برنامه های اینترنت اشیا تبدیل شده است. با این حال، از آنجایی که تعداد زیادی از برنامه های اینترنت اشیا نیاز به اجرا در منابع لبه / مه دارند، ممکن است سرورها بیش از حد بارگذاری شوند. از این رو، ممکن است سرورهای لبه / مه را مختل کند و همچنین بر زمان پاسخگویی برنامه های اینترنت اشیا تأثیر منفی بگذارد. علاوه بر این، بسیاری از برنامه های کاربردی اینترنت اشیا از اجزای وابسته تشکیل شده اند که محدودیت های اضافی را برای اجرای خود متحمل می شوند. علاوه بر این، محیط های محاسباتی لبه / مه و برنامه های اینترنت اشیا ذاتاً پویا و تصادفی هستند. بنابراین، زمان بندی کارآمد و سازگار برنامه های اینرنت اشیا در محیط های محاسباتی لبه / مه ناهمگن از اهمیت بالایی برخوردار است. با این حال، منابع محاسباتی محدود در سرورهای لبه / مه بار اضافی را برای اعمال تکنیک های بهینه اما محاسباتی نیازمند تحمیل می کند. برای غلبه بر این چالش ها، و انگ و همکارانش[2] یک الگوریتم زمانبندی مبتنی بر یادگیری تقویتی را پیشنهاد کردند که تا زمان پاسخ دهی برنامه های ناهمگن اینترنت اشیا را به طور سازگار و کارآمد بهینه کند و بار سرورهای لبه / مه را متعادل کند. نتایج به دست آمده از آزمایش های گسترده نشان می دهد که روش پیشنهادی در مقایسه با الگوریتم های فراابتکاری و سایر تقویت کننده ها، به ترتیب تا ۵۵، ۳۷ درصد و ۵۰ درصد هزینه اجرای برنامه های ابر و مه را از نظر تعادل بار، زمان پاسخ و هزینه وزنی کاهش می دهد.
* در مقایسه با ابر، محاسبات مه به عنوان یک سیستم توزیع شده متشکل از میلیون ها دستگاه که در حداقل فاصله از دستگاه های مشتری قرار دارند، حیاتی است. علاوه بر این، زیرساخت مه باعث کاهش پهنای باند و تأخیر می شود زیرا به کاربر نهایی نزدیک تر است. با این حال، به حداکثر رساندن استفاده از منابع، به حداقل رساندن زمان پاسخ، و اطمینان از تکمیل وظایف محدود به ضرب الاجل در مهلت مقرر، از مشکلات تحقیقاتی مهم در محاسبات مه هستند. هوسین و همکارانش[21] یک تکنیک زمان بندی کار به نام زمان بندی وظایف اولویت دار آگاه از منابع را در یک محیط محاسباتی مه ناهمگن پیشنهاد کردند. هدف پژوهش این است که وظایف محدود به ضرب الاجل را در مهلت های زمانی خود اجرا کرده، زمان پاسخگویی و هزینه را به حداقل برسانند، و همچنین زمان ساخت را به حداکثر برسانند و استفاده از منابع لایه مه را به حداکثر برسانند. روش پیشنهادی با استفاده از iFogSim پیاده سازی شد و عملکرد آن از نظر زمان پاسخ، استفاده از منابع، ضرب الاجل های کار، هزینه، و طول عمر ارزیابی شد. نتایج با زمان بندی های پیشرفته مه مانند RACE (CFP) و RACE (FOP) مقایسه شده اند. نتایج نشان می دهد که روش پیشنهادی منابع، زمان پاسخ دهی، زمان ساخت، هزینه و انجام مهلت های کار را بهبود داده است.
* چافی و همکارانش[5] یک تحلیل مقایسه ای از الگوریتم های زمان بندی مه و محاسبات لبه را ارائه می کنند که با تمرکز بر تأثیر آن ها بر بهینه سازی حجم کار. تجزیه و تحلیل تخصیص وظیفه، تعادل بار، بهره وری انرژی، تأخیر و مقیاس پذیری، زمانبندی بهینه در نظر می گیرند. در این پژوهشالگوریتم های موجود، در کنار اصلاحات و مقدمه های جدید، ارزیابی شدند. برنامه های کاربردی دنیای واقعی ملاحظات عملی و سازگاری این الگوریتم ها را به نمایش می گذارند. معیارهای تصفیه شده اثرات فوری و بلندمدت را بر مقیاس پذیری نشان می دهند. نتایج به تحقیقات مه و محاسبات لبه آینده، با در نظر گرفتن فناوری های نوظهور کمک می کند. این مطالعه عوامل کلیدی را برای بهینه سازی حجم کاری اینترنت اشیا شناسایی می کند و اهمیت الگوریتم های زمان بندی را برای به حداکثر رساندن عملکرد برجسته می کند.
* هدف این مقاله بهبود زمان پردازش کار کلی برنامه های کاربردی اینترنت اشیا (IoT) تلفن همراه در محاسبات ابری مه، با در نظر گرفتن منابع مختلف و الزامات امنیتی همراه با محدودیت های زمانی کار است. محاسبات مه منابع ابری را برای سرویس دهی به دستگاه های اینترنت اشیا در لبه شبکه گسترش می دهد. در چنین سناریویی، تصمیم گیری در مورد اینکه آیا وظایف باید در لایه مه پردازش شوند یا به ابر ارسال شوند بسیار مهم است. علاوه بر این، برای کاربردهای بلادرنگ، تحرک دستگاه های IoT و پهنای باند محدود موجود در دستگاه های لبه، زمان پردازش پایین کار را به خطر می اندازد. علاوه بر این، الزامات امنیتی برخی از برنامه های کاربردی اینترنت اشیا (به عنوان مثال مراقبت های بهداشتی) نیازمند پردازش وظایف توسط سرورهای مه یا ابر خاص برای اطمینان از محرمانه بودن اطلاعات است، که ممکن است زمان پردازش کار را نیز به تأخیر بیندازد. بنابراین، ما ابتدا با پیشنهاد سه الگوریتم مختلف که بر روی تخصیص دستگاه IoT موبایل به دستگاه لبه مناسب (به عنوان مثال دروازه مه) با در نظر گرفتن فاکتورهای بار فاصله و پهنای باند کار می کنند، به موضوع تحرک می پردازیم. سپس، یک الگوریتم زمان بندی کار جدید را ارائه می دهیم که از منطق فازی برای بهینه سازی توزیع وظایف بین لایه های مه و ابر، با در نظر گرفتن الزامات امنیتی کار، استفاده می کند. الگوریتم واحد پردازش مناسب را برای اجرای وظیفه در لایه مه با بهره برداری از خواسته های وظیفه (مثلاً محاسبات، ذخیره سازی، پهنای باند، امنیت) و مهلت انتخاب می کند. نتایج نشان می دهد که در نظر گرفتن فاکتورهای فاصله و بار پهنای باند موجود هنگام تخصیص دستگاه IoT به دروازه مه، زمان پردازش کار را بهتر از اتخاذ یک جنبه بهبود می بخشد. نتایج همچنین نشان می دهد که الگوریتم زمان بندی پیشنهادی ما از سایر الگوریتم های موجود در رابطه با زمان ساخت، زمان چرخش، نسبت موفقیت و معیارهای زمان پردازش بهتر عمل می کند.
* محاسبات داوطلبانه مه یک محاسبات توزیع شده مبتنی بر اینترنت است که در آن داوطلبان منابع اضافی موجود خود را برای مدیریت وظایف در مقیاس بزرگ به اشتراک می گذارند. با این حال، دستگاه های محاسباتی در یک سیستم محاسبات داوطلبانه از نظر قدرت پردازش، هزینه پولی و تأخیر انتقال داده بسیار پویا و ناهمگن هستند. برای اطمینان از کیفیت بالای خدمات و هزینه کم برای درخواست های مختلف، همه منابع محاسباتی موجود باید به طور کارآمد استفاده شوند. از این رو استفاده از فرایندهای جای گذاری سرویس در این ساختارها، اهمیت بالایی دارد. جای گذاری سرویس وظایف یک مسئله NP-hard است که به عنوان یکی از چالش های اصلی در یک سیستم محاسبات مه داوطلبانه ناهمگن در نظر گرفته می شود. با توجه به این موضوع، گروه تحقیقاتی حسینی و همکارانش[22] زمان بندی وظایف برای محاسبات مه داوطلبانه به نام های Min-CCV و Min-V را طراحی کردند. هدف اصلی الگوریتم های پیشنهادی به طور مشترک به حداقل رساندن هزینه های محاسباتی، ارتباطی و تأخیر تخلف برای درخواست های کاربران د محاسبات ابری است. نتایج شبیه سازی گسترده نشان می دهد که الگوریتم های پیشنهادی قادر به تخصیص وظایف به منابع داوطلبانه مه و ابر به طور مؤثرتری نسبت به حالت های پیشرفته هستند. به طور خاص، الگوریتم های پیشنهادی در مقایسه با الگوریتم مبتنی بر ژنتیک، میزان رضایت را حدود 99.5 درصد بهبود می بخشند و هزینه کل را بین 15 تا 53 درصد کاهش می دهند.
* گروه تحقیقاتی رینو و همکارانش[3] یک الگوریتم زمانبندی وظایف جدید در محاسبات مه-ابر به نام برای بهینه سازی تابع چند هدفه پیشنهاد می کنند که شامل یک ساختار از مجموع وزنی زمان محاسبات کلی، مصرف انرژی و درصد وظایف برآورده شده در مهلت زمانی مشخص است. در این پژوهش نیازهای مختلف وظایف و ماهیت ناهمگون گره های مه و ابر در نظر گرفته می شود. در این پژوهش یک رویکرد ترکیبی مبتنی بر اولویت بندی وظایف و یک الگوریتم ژنتیک برای یافتن یک گره محاسباتی ترجیحی برای هر کار پیشنهاد می شود. شبیه سازی های گسترده، الگوریتم پیشنهادی را برای نشان دادن برتری آن نسبت به استراتژی های پیشرفته ارزیابی می کنند.
* گروه تحقیقاتی لاخان و همکارانش[23] یک چارچوب الگوریتم زمانبندی وظایف وظایف کار آمد با در نظر گرفتن پارامترهای انرژی، بر اساس شبکه های عصبی عمیق را پیشنهاد کردند که شامل بخش های پارتیشن بندی برنامه، توالی کار، و زمان بندی هست. نتایج تجربی روش های پیشنهادی را از نظر مصرف انرژی و هزینه کاربرد در محیط پویا نشان می دهد.
* گروه تحقیقاتی لیجز و همکارانش[4]، فرایند زمانبندی وظایف گردش کار را در محیط های مه-ابر بررسی کردند تا یک رویکرد زمانبدی بهینه وظایف با در نظر گرفتن مسئله مصرف انرژی در زمان تکمیل برنامه ارائه کنند. در این پژوهش یک الگوریتم جای گذاری سرویس، بهینه سازی چند هدفه ارائه می شود که در دو فاز کار می کند. این رویکرد، مسئله را به عنوان یک مسئله بهینه سازی چند هدفه مدل سازی می کند و هنگام تخصیص منابع مه و ابر، مبادله بین اهداف متضاد را محاسبه می کند، و وظایف حساس به تأخیر (با نیازهای محاسباتی کمتر) را برای منابع و وظایف پیچیده محاسباتی (با تأخیر کم) جای گذاری سرویس می کند. همچنین در روش پیشنهادی رویکرد مقیاس بندی فرکانس گام به گام آگاه از سر رسید زمانی برای کاهش بیشتر مصرف انرژی با استفاده از شکاف های زمانی استفاده نشده بین دو وظیفه از قبل برنامه ریزی شده در یک گره تطبیق داده می شود. نتایج ارزیابی با استفاده از برنامه های کاربردی سنتز شده و دنیای واقعی نشان می دهد که رویکرد پیشنهادی گروه تحقیقاتی لیجز و همکارانش مصرف انرژی را تا 50٪ در مقایسه با رویکردهای موجود با حداقل تأثیر بر زمان تکمیل کاهش می دهد.
* گروه تحقیقاتی معماری و همکارانش [11] یک الگوریتم زمانبندی وظایف آگاه از تأخیر مبتنی بر تطبیق ماشین مجازی با استفاده از رویکردهای فراابتکاری را ارائه کردند. در میان روش های اکتشافی، جستجوی Tabu به دلیل گسترش زیاد آن در مسائل مختلف بهینه سازی و همچنین ویژگی های حافظه و سرعت بالا، آن را به یک روش معمول تبدیل می کند. بنابراین، در این پژوهش یک الگوریتم جدید مبتنی بر جستجوی Tabu پیشنهاد شد که با استفاده از الگوریتم های تقریبی نزدیک ترین همسایه و بهینه سازی مگس میوه بهبود می یابد. در نهایت، برای اعتبارسنجی روش پیشنهادی، یک مطالعه موردی شبیه سازی شده و الگوریتم پیشنهادی با در نظر گرفتن پارامترهای زمان اجرا، تأخیر، حافظه تخصیص یافته و تابع هزینه پیاده سازی می شود. نتایج مقایسه نشان می دهد که الگوریتم پیشنهادی از روش های جستجوی Tabu، الگوریتم ژنتیک، PSO و تبرید شبیه سازی شده بهتر عمل می کند.
* گروه تحقیقاتی رفعت و همکارانش[12] یک راه حل زمانبندی وظایف را پیشنهاد می کنند که معماری محاسبات مه را به منظور برآورده کردن حداکثر تعداد درخواست ها با توجه به الزامات مهلت زمانی آنها اتخاذ می کند. در این کار، یک مدل بهینه سازی با استفاده از برنامه نویسی اعداد صحیح مختلط برای به حداقل رساندن مسائل در مهلت زمانی معرفی شده است. مدل پیشنهادی با یک تکنیک دقیق اعتبار سنجی می شود. در این راستا با توجه به ماهیت پیچیده مسئله، یک رویکرد اکتشافی با استفاده از الگوریتم ژنتیک (GA) ارائه شده است. در این راستا عملکرد الگوریتم ژنتیک پیشنهادی با جای گذاری سرویس دوره ای و اولویت بندی ارزیابی و مقایسه شده است و نتایج نشان می دهد که بهبود مهلت زمانی روش پیشنهادی 20٪ تا 55٪ بهتر از تکنیک های دیگر است.
* گروه تحقیقاتی جیا و همکارانش[8] در پژوهش خود، مسئله بهینه سازی زمانبندی وظایف محاسباتی را با منابع ناهمگن در یک سیستم محاسباتی مه-ابر در نظر گرفتند. این مسئله در بسیاری از برنامه های بی درنگ و سیار رایج است، جایی که وظایف عظیم و محاسباتی فشرده هستند، و منابع محاسباتی می توانند هم دستگاه های محاسبات مه و هم پلتفرم های ابری را شامل شوند. چالش ها در پیشنهاد الگوریتم های مؤثر، کارآمد و قوی با هدف به حداقل رساندن تأخیر کل و مصرف انرژی نهفته است. در پژوهش گروه تحقیقاتی جیا و همکارانش یک مدل جای گذاری سرویس کار دوهدفه فرموله شده است، که در آن مدل های صف برای تخمین تأخیر و مدل های مصرف انرژی برای منابع ناهمگن معرفی می شوند. یک چارچوب جای گذاری سرویس وظایف عظیم مبتنی بر بهینه سازی جبهه پارتو برای جای گذاری سرویس وظایف عظیم در یک واحد زمانی پیشنهاد شده است. این رویکرد از یک مجموعه راه حل غیر نظارتی به دست آمده از روش های جستجوی محلی آگاه از تأخیر انتقال شروع می شود. یک روش جستجوی محلی مبتنی بر درخت برای بهبود بیشتر راه حل های غیر نظارتی پیشنهاد شده است. الگوریتم پیشنهادی با چهار الگوریتم کلاسیک برای مسائل مشابه مقایسه شده است. عملکرد روش های پیشنهادی با معیارهای بهینه سازی پارتو در جنبه های مختلف ارزیابی می شود. نتایج تجربی اثربخشی و استحکام پیشنهاد را برای مسئله مورد مطالعه نشان می دهد.
* گروه تحقیقاتی دوبی و همکارانش[24] از ویژگی های دو تکنیک مبتنی بر فراابتکاری بهینه سازی جستجوی فاخته و بهینه سازی ازدحام ذرات بهره برداری می کنند. در این پژوهش یک چارچوب امن برای حل تخصیص خدمات اینترنت اشیا در محیط ابری-مه ایجاد پیشنهاد می شود و در عین حال پارامترهای تأثیرگذار ذکر شده را به حداقل می رسد. عملکرد چارچوب پیشنهادی به دقت در مجموعه داده های مصنوعی و ناهمگونی منابع در مه و همچنین محیط شبیه سازی ابر ارزیابی می شود. نتایج شبیه سازی ثابت کرد که الگوریتم فراابتکاری ترکیبی پیشنهادی بهتر از سایر سیاست های پایه عمل می کند و پارامترهای تأثیرگذار مختلف را بهبود می بخشد.
* گروه تحقیقاتی موکنی و همکارانش[14]، یک رویکرد چند عاملی ترکیبی مبتنی بر ابر و مه را برای زمانبندی وظایف مجموعه ای از وظایف وابسته که به عنوان یک گردش کار مدل سازی شده اند، پیشنهاد کردند. مزیت اصلی رویکرد پیشنهادی این است که امکان مدل سازی جای گذاری سرویس گردش کار اینترنت اشیا به عنوان یک مسئله بهینه سازی چند هدفی به منظور ایجاد یک راه حل جای گذاری سرویس سازش آمیز از نظر زمان پاسخ گویی، هزینه و طول عمر را فراهم می کند. علاوه بر در نظر گرفتن ارتباطات داده ای بین وظایف گردش کار، در طول فرآیند برنامه ریزی، رویکرد پیشنهادی شامل دو مزیت به حداکثر رساندن استفاده از محاسبات مه به منظور به حداقل رساندن زمان پاسخ و استفاده از محاسبات ابری الاستیک هست. مجموعه آزمایش های انجام شده بر روی سناریوهای مختلف و همچنین تجزیه و تحلیل نتایج یافت شده، امکان سنجی رویکرد پیشنهادی و عملکرد آن را از نظر هزینه تأیید می کند که نشان دهنده میانگین سود 21.38٪ در مقایسه رویکردهای اریابی شده است. همچنین زمان اجرا کاهش 14.13 درصدی نسبت به رویکردهای مقایسه شده را نشان می دهد.
* گروه تحقیقاتی بیشت و همکارانش[25] یک نسخه بهبود یافته از الگوریتم min-min برای زمانبندی و جای گذاری سرویس گردش کار پیشنهاد کرده اند که هزینه، طول عمر، انرژی و تعادل بار را در محیط ناهمگن در نظر می گیرد. این الگوریتم در سناریوهای مختلف بارگذاری فقط ابر، فقط مه و همکاری بین ابر و مه پیاده سازی و آزمایش می شود. روش پیشنهادی عملکرد بالایی در محیط ابر و مه ارائه کرده و در مقایسه با الگوریتم های min-min و ELBMM حداقل طول عمر، مصرف انرژی کمتر همراه با متعادل سازی بار و هزینه ی اندکی کمتر را ارائه می دهد.
* گروه تحقیقاتی اوکک بیله و همکارانش[26] یک چارچوب زمانبندی وظایف مبتنی بر محاسبات مه چند کاربره، چند کلاسه و چند لایه برای اجرای مؤثر وظایف و فرآیندهای محاسباتی ارائه کردند. الزامات مهم سیستمی که در راه حل های چندلایه موجود مانند بارگیری، محاسبات و الزامات مهلت زمانی در مدل سازی سیستم ثبت نشده اند، در روش پیشنهادی در نظر گرفته شده است. در این پژوهش سیستم سه لایه در نظر گرفته شد که در آن هر دستگاه وظایف تولید شده خود را در هر شکاف زمانی به هر لایه انتخابی برای محاسبه تخصیص می دهد. پس از ورود به چنین لایه انتخابی، وظیفه تنها در صورتی پذیرفته می شود که اندازه صف کمتر از آستانه از پیش تعریف شده باشد، در غیر این صورت، چنین وظایفی به لایه بعدی بارگذاری می شود. وظایف بر اساس الزامات کیفیت خدمات به وظایف کلاس 1 و کلاس 2 طبقه بندی شدند. در این پژوهش تکنیک های هندسه تصادفی، محاسبات موازی و تئوری صف برای مدل سازی عملکرد محیط محاسباتی یکپارچه لبه-مه-ابر در نظر گرفته شد و تجزیه و تحلیل را برای معیارهای عملکرد مختلف به دست آمد. تحلیل های به دست آمده اهمیت سیستم های محاسبات لبه ای چند لایه و چند کلاسه را برای بهبود تجربه برنامه های کاربردی حساس به تأخیر و حیاتی در هر محیط تخلیه کار نشان می دهند.
* گروه تحقیقاتی موکنی و همکارانش[27] رویکرد مبتنی بر الگوریتم ژنتیک و سیستم چند عامله را برای زمانبندی و جای گذاری سرویس گردش وظایف در محاسبات ابر و مه ارائه کردند که هدف آن بهینه سازی مقادیر زمان، هزینه و تأخیر است. رویکرد پیشنهادی بر اساس سیستم چند عامله است، با در نظر گرفتن معیارهای مختلف کیفیت خدمات. یک روش بهینه سازی چند هدفه برای ایجاد راه حل جای گذاری سرویس گردش کار با توسعه الگوریتم ژنتیک اتخاذ شده است. این فصل یک سیستم چند عامله را ارائه می کند که شامل مجموعه ای از انواع عامل با یک مأموریت خاص برای ملاقات است. الگوریتم ژنتیک یکی از رویکردهای فراابتکاری های پر استفاده برای حل مسائل بهینه سازی چندهدفه با بهینه سازی معیارهای کیفیت سرویس راه حل زمان بندی گردش کار است. نتایج تجربی نشان می دهد که رویکرد پیشنهادی برای جای گذاری سرویس گردش وظایف اینترنت اشیا در محاسبات ابر و مه یک راه حل جای گذاری سرویس ایجاد می کند که مقادیر هزینه و تأخیر را هنگام به حداکثر رساندن تعداد کار گردش کار در منابع ابر بهینه می کند.
* گروه تحقیقاتی جیان و همکارانش[28]، روش کاملاً جدیدی را پیشنهاد می کنند که می تواند زمان و انرژی مصرفی ماشین مورد استفاده برای حل مشکلات ذکر شده در بالا را کاهش دهد. ابتدا با توجه به عملکرد عالی الگوریتم ازدحام خفاش ها در مسئله NP-hard، دومین عامل انعطاف پذیری برای جلوگیری از همگرایی زودرس معرفی می شود و فناوری فضا برای جستجوی سفارش مبادله و مهاجرت ترکیب می شود. این گروه تحقیقاتی یک الگوریتم OEMBA ایجاد کردند که سرورهایی را که اغلب برای صرفه جویی در انرژی استفاده نمی شوند، جمع می کند. سپس، مدل بهبود یافته حافظه کوتاه مدت برای تثبیت مکان VM و کاهش تأخیر بر اساس داده های تاریخی استفاده می شود. نتایج نشان می دهد که روش یادگیری بهبودیافته می تواند باعث صرفه جویی در انرژی و کاهش تأخیر توزیع شود. بر اساس ویژگی های محیط وسیع تر، روش یادگیری توزیع شده کارآمد ارائه شده در این مقاله می تواند صرفه جویی در انرژی و تأخیر کم را در نظر بگیرد. استفاده از داده های تاریخی برای آموزش یک مدل یادگیری نیز ایده جدیدی برای تحقیقات VMP ارائه می دهد. آزمایش های شبیه سازی بر روی پلتفرم EdgeCloudSim انجام شد و نتایج نشان می دهد که در مقایسه با BA، GA و PSO، الگوریتم OEMBA می تواند به طور مؤثر مصرف انرژی سرور لبه را کاهش دهد و تأثیر مثبتی بر افزایش تعداد سرورها داشته باشد. روش پیشرفته یادگیری ماشین (LSTM) نه تنها می تواند به طور مؤثر نتایج توزیع را پیش بینی کند، بلکه سرعت توزیع را نیز بهبود می بخشد و تأخیر مورد نیاز برای مکان های دور را کاهش می دهد. با این حال، کار پیشنهادی این گروه تحقیقاتی به درستی محیط ابر واقعی را شبیه سازی نکرده و نرخ ورود شغل، تعداد درخواست های شغلی در ثانیه و غیره را در نظر نگرفته است. الزامات خدمات متفاوت است و برخی از خدمات به نام خاصی نیاز دارند.
* ترکیب محاسبات ابری با اینترنت اشیا تغییرات اساسی در همه چیز از صنعت، مراقبت های بهداشتی، ترافیک و حمل و نقل گرفته تا لوازم خانگی و حتی زندگی شخصی ایجاد کرده است. میلیاردها دستگاه و کاربر از طریق این پلتفرم ها به هم متصل می شوند و حجم زیادی از داده ها را در معرض دید قرار می دهند که عملکرد را کاهش می دهند، که تقاضا برای برنامه های استقرار از پیش برنامه ریزی شده را افزایش داده است. گروه تحقیقاتی صراف زاده و همکارانش[29] یک الگوریتم زمانبندی مبتنی بر ژنتیک در یک محیط ابری را با تمرکز بر کاهش تأخیر پروژه و استفاده از شبکه پیشنهاد کردند. در این کار، یک رویکرد مبتنی بر جریمه برای هدف قرار دادن تأخیر و تعداد زمان بر استقرار ابر معرفی شده است، که راه حل ها را به عنوان تابعی از تولید بررسی می کند. این کمک می کند تا در ابتدا یک منطقه بزرگ تر کشف شود و به تدریج تأثیر مجازات در نسل های بعدی عمیق تر شود. در یک سطح جداگانه، نزدیکی اپلیکیشن ها به کاربران نیز در نظر گرفته شده است. این کار از طریق یک فرآیند انتخاب کروموزوم با استفاده از یک مقدار اولویت انجام می شود که مجاورت ماژول های وابسته را مشخص می کند. نتایج شبیه سازی نشان می دهد که الگوریتم پیشنهادی تأخیر، استفاده از شبکه، مصرف انرژی و هزینه را بهبود بخشیده است.
* بر اساس پژوهش گروه تحقیقاتی تقی زاده و همکارانش[30]اطلاعات مربوط به تمرین اینترنت اشیا قابل پردازش و ذخیره در گره های LFOG و RFOG می باشد. به طور کلی داده های اینترنت اشیا هر روز در حال رشد و افزایش است و داده های بیشتری برای پردازش به رایانش ابری یا گره های ابری منتقل می شود. یکپارچه سازی اطلاعات در مورد دستگاه های اینترنت اشیا نزدیک به یکی از چالش های اصلی در این زمینه تبدیل شده است و توجه ویژه ای را می طلبد. برای بهبود در دسترس بودن داده ها و کاهش تأخیر و هزینه اکتساب داده، آنها از متاالگوریتم NSGA-II برای توزیع داده ها در بهترین قسمت مه نزدیک به کاربر استفاده کردند. همچنین چارچوبی برای هدایت ادغام منابع داده در زیرساخت رایانش ابری فراهم می کند. مدل شبیه سازی در شبیه ساز iFogSim تحت شرایط مشابه دو روش اصلی دیگر با استفاده از متغیرهایی مانند کپی ها، تعداد کاربران و گره های ابری اعتبارسنجی می شود. نتایج نشان می دهد که روش تجربی از نظر در دسترس بودن داده ها، چهار تأخیر اطلاعاتی و هزینه اطلاعات بهتر از دو روش اصلی دیگر کار می کند. برای بهبود بهره وری انرژی، کار آینده ما بر ایمنی و کاهش مصرف انرژی متمرکز خواهد شد. بهره وری انرژی را می توان با کاهش مصرف انرژی با تنظیم موقعیت گره های داده بر اساس بار به دست آورد. برنامه های کاربردی اینترنت اشیا اغلب برای جمع آوری و مدیریت داده های تولید شده توسط دستگاه های اینترنت اشیا مانند لوازم خانگی، دستگاه ها و غیره استفاده می شوند. گوشی های هوشمند و غیره. از آنجایی که حجم داده های تولید شده بسیار زیاد است، ما آنها را برنامه های کاربردی اینترنت اشیا مبتنی بر داده می نامیم. برخی از برنامه های اینترنت اشیا که از داده ها استفاده می کنند شامل برنامه های نظارتی (مانند دوربین های نظارتی)، برنامه های کاربردی شهر هوشمند (مانند کنترل ترافیک شهر)، برنامه های کاربردی مراقبت های بهداشتی (مانند پوشیدنی ها) و نرم افزارهای تلفن همراه (مانند مدیریت) هستند. صفر ربات ها در تولید). برخی از این برنامه ها با چالش هایی مانند امنیت و حریم خصوصی، ناهمگونی و QoS جریان داده اینترنت اشیا مواجه هستند. بنابراین، مدیریت داده های IoT قدرت اطلاعاتی می تواند نقش مهمی در بهبود کارایی استفاده از رایانش ابری داشته باشد. این یکی از سخت ترین شغل هایی است که می توان به آن فکر کرد. 6،7 در دنیای اینترنت اشیا، ما از داده های توزیع شده انبوه استفاده می کنیم و برنامه های فشرده داده را در گره های ابری مستقر می کنیم. برای جلوگیری از مشکلات تماس و افزایش دسترسی، مه های با استحکام کافی نیز توصیه می شود. پردازش داده ها یک مشکل NP-hard است که می تواند در مدت زمان معقولی با استفاده از روش های اکتشافی حل شود. این روش ها به میزان ظرفیت مه و توزیع جغرافیایی در محیط مه بستگی دارد.
* به عنوان یکی از موضوعات باز اصلی، مشکل مکان سرویس می تواند نقش مؤثری در حوزه مدیریت منابع داشته باشد و تأثیر به سزایی در استفاده از پهنای باند و تأخیر شبکه های IoT دارد. اگرچه چندین مکانیسم ارائه خدمات برای یافتن راه حل های ارائه خدمات غیر بهینه مورد بررسی قرار گرفته اند، هنوز کار بیشتری برای استقرار خودکار خدمات اینترنت اشیا بهتر مورد نیاز است. گروه تحقیقاتی آرانی و همکارانش[31] راه حلی کارآمد برای توزیع خدمات اینترنت اشیا با الهام از استفاده از یک مدل محاسباتی مستقل برای استقرار برنامه های کاربردی اینترنت اشیا در زیرساخت ابر مجازی ارائه می دهند. راه حل ما از مدل ارائه شده توسط IBM پیروی می کند که شامل چهار مرحله نظارت، تجزیه و تحلیل، برنامه ریزی و اجرا با یک پایگاه دانش مشترک به نام MAPE-k ring برای توزیع خدمات IBM به گره های مه است. الزامات QoS آنها به طور خودکار. اول، اطلاعات لازم در مورد الزامات QoS خدمات اینترنت اشیا و قابلیت های گره های مه موجود به طور مداوم در طول دوره نظارت جمع آوری می شود. اطلاعات جمع آوری شده برای اولویت بندی خدمات اینترنت اشیا بر اساس مهلت ها و زمان انتظار استفاده می شود. سپس در مرحله برنامه ریزی، یک راه حل مناسب با استفاده از الگوریتم بهینه سازی نهنگ فراابتکاری (WOA) تعیین می شود. ساختار مدل پیشنهادی در شکل زیر نشان داده شده است. در این مطالعه، محققان به مشکل ارائه خدمات با استفاده از یک روش خودکار برای استقرار برنامه های IoT در زیرساخت ابر مجازی پرداختند. با توجه به تنوع نیازمندی های QoS خدمات اینترنت اشیا و ناهمگونی منابع در اکوسیستم مه و گره های مه با محدودیت منابع، یافتن راه حل ارائه خدمات کارآمد به عنوان یکی از چالش ها مهم است. محققان از روش فراابتکاری WOA برای استقرار مؤثر برنامه های IoT بر روی گره های مه موجود استفاده کردند. همچنین، یک چارچوب ارائه خدمات مستقل با توجه به معماری سه لایه اکوسیستم مه توسعه یافته است تا تعامل بین اجزای اصلی و لایه های مه در دستگاه اینترنت اشیا را نشان دهد. رویکرد پیشنهادی تحت سرویس های اینترنت اشیا با الزامات QoS مختلف و گره های مه با قابلیت های منابع مختلف آزمایش شده است. نتایج به دست آمده نشان می دهد که از نظر بهره برداری از منابع و نسبت پذیرش خدمات، مصرف انرژی و تأخیر خدمات، نسبت به سایر مکانیسم های مبتنی بر فراابتکاری برتری دارد. برای کارهای آینده، محققان قصد دارند مکانیسم پیشنهادی را با سیستم های مبتنی بر بلاک چین گسترش دهند تا از حریم خصوصی بین برنامه های IoT اطمینان حاصل کنند و مقیاس پذیری رویکرد پیشنهادی را در یک برنامه استاندارد Kubernetes آزمایش کنند. علاوه بر این، یک الگوریتم ترکیبی با استفاده از بهینه ساز گرگ خاکستری (GWO) و بازپخت شبیه سازی شده (SA) برای ارائه یک راه حل خدمات مؤثر، و همچنین یک روش مبتنی بر تأیید رسمی برای اثبات صحت و تأیید مدل توسعه خواهد یافت. مهارت های رفتاری.
* در حال حاضر مدیریت منابع یکی از وظایف اصلی در تحقیقات مبتنی بر محاسبات مه است. مسائل زیادی در رابطه با مدیریت منابع ابری و محاسبات مه وجود دارد، مانند پیش بینی منابع، تأمین منابع، تأمین خدمات، برنامه ریزی سرویس، استقرار و مهاجرت سرویس. یکی از بزرگ ترین چالش ها هنگام پیاده سازی محاسبات مه «نحوه استقرار مؤثر خدمات گره مه» است. زیرا برخلاف مراکز داده ابری، سخت افزار ابر پویا دارای منابع محدود و توزیع شده ای است که مدیریت منابع را دشوار می کند. بنابراین، یکی از مهم ترین مشکلات در مدیریت منابع مه و ابر، تصمیم به ارائه خدمات یا خدمات میزبانی توسط خودتان است. گروه تحقیقاتی تحقیقاتی سالیمان و همکارانش[32] یک مدل ریاضی مستقل مبتنی بر مدیریت متمرکز ابرهای مه برای مدیریت مناسب ترین منابع مه و ابر ارائه کردند. هدف از حل این مشکل، استقرار مؤلفه های توزیع شده برنامه های IoT (به عنوان مثال، خدمات) در منابع مه و ابر است که امکان نقشه برداری بین گره های مه و برنامه های IoT را برای استفاده از منابع فراهم می کند. مزایای ارائه خدمات پویا شامل کاهش مصرف پهنای باند، مدیریت بهتر منابع، کاهش هزینه های محاسباتی، در دسترس بودن، کاهش تأخیر، افزایش قابلیت اطمینان، کاهش مصرف انرژی، بهبود کیفیت خدمات و برآوردن نیازهای زمان پاسخگویی برنامه های کاربردی به موقع است. شرکت ماشین های تجاری بین المللی (IBM) یک مدل MADE-k مستقل برای پیاده سازی محاسبات مه معرفی کرده است. این مدل از چهار مرحله نظارت، تجزیه و تحلیل، تصمیم گیری و اجرا تشکیل شده است که یک پایگاه دانش مشترک را تشکیل می دهد. خدمات اینترنت اشیا و منابع مه در مرحله نظارت کنترل می شوند. مرحله تجزیه و تحلیل وظیفه اولویت بندی اجرای خدمات بسته به زمان را بر عهده دارد. در مرحله تصمیم گیری، برنامه ریزی برای استقرار مناسب خدمات انجام می شود و در نهایت در مرحله اجرا، استقرار برنامه ریزی شده اعمال می شود.
* • گروه تحقیقاتی تکیه بند و همکاران [33] مشکل ارائه خدمات پویا را برای ارائه خدمات کاربردی IoT در زیرساخت مه ناهمگن و پویا مورد مطالعه قرار داد. از آنجایی که رفتار مصرف منابع در برنامه های خاص اینترنت اشیا در طول زمان تغییر می کند، ارائه پویا خدمات باید به عنوان یکی از مسائل چالش برانگیز در مدیریت برنامه های اینترنت اشیا در نظر گرفته شود. محققان از اتومات ای یادگیری برای شناسایی تصمیمات ارائه خدمات برای استقرار یا استقرار برنامه های کاربردی اینترنت اشیا در زیرساخت ابر پویا استفاده کردند. آنها مکانیسم پیشنهادی را تحت ردیابی ترافیک شبیه سازی شده و ردیابی ترافیک در دنیای واقعی ارزیابی کردند و نتایج نشان داد که در مقایسه با مکانیسم های دیگر، تأخیر سرویس، هزینه و تخلف خدمات را کاهش می دهد. بنابراین مشکل ارائه خدمات پویا به عنوان یکی از وظایف چالش برانگیز می تواند نقش مهمی در مدیریت مؤثر اپلیکیشن ایفا کند و تأثیر آن بر تأخیر مورد نیاز اپلیکیشن های اینترنت اشیا و استفاده از پهنای باند شبکه های مه قابل توجه است. اگرچه برخی از استراتژی های ارائه خدمات پویا قبلاً برای دستیابی به پویاترین راه حل های ارائه خدمات بررسی شده اند، تلاش های بیشتری برای ارائه بهتر و خودکار خدمات اینترنت اشیا مورد نیاز است. در این مقاله، یک مدیر تحویل خدمات پویا مستقل (DSPM) ارائه می کنیم که از یک حلقه کنترل خود مدیریتی پیروی می کند. فعال کردن برنامه های IoT در زیرساخت ابر DSPM از سه جزء فرعی تشکیل شده است که عبارتند از: مانیتورینگ، تحلیلگر حجم کار و ارائه دهنده خدمات. در مرحله اول، مؤلفه مانیتورینگ اطلاعات درخواست های دریافتی از کاربران اینترنت اشیا، الزامات QoS برنامه های اینترنت اشیا و قابلیت های منابع گره های مه را از اکوسیستم مه ثبت می کند. سپس، مؤلفه تحلیل حجم کار، وضعیت آینده سیستم را بر اساس ترافیک به گره های مه با استفاده از مدل پیش بینی سری زمانی تخمین می زند. در نهایت، ارائه دهنده خدمات از تکنیک های مبتنی بر یادگیری ماشین برای تعیین خودکار تصمیم های ارائه خدمات برای استقرار یا استقرار برنامه های IoT استفاده می کند. این مراحل به طور مداوم در فواصل زمانی از پیش تعیین شده انجام می شود و خدمات اینترنت اشیا را به گره های مه موجود با توجه به نیازهای QoS آنها ارائه می دهد.
* توسعه سریع برنامه های کاربردی مبتنی بر عصر اینترنت اشیا و شبکه های نسل پنجم منجر به افزایش تصاعدی در میزان داده های مورد نیاز برای پردازش خدمات اینترنت اشیا شده است. مدل محاسبات مه به عنوان یک راه حل محاسباتی توزیع شده برای خدمت به این برنامه ها با استفاده از گره های مه واقع در نزدیکی دستگاه های IoT ظهور کرده است. از آنجایی که برنامه های اینترنت اشیا در اکوسیستم مه به عنوان چندین سرویس اینترنت اشیا با نیازهای خدماتی با کیفیت متفاوت توسعه می یابند که می توانند روی گره های مه با ظرفیت های منابع مختلف مستقر شوند، یافتن یک طرح استقرار سرویس کارآمد بسیار دشوار است. مسائلی که باید در نظر گرفته شود. گروه تحقیقاتی آرانی و همکارانش [31]، یک راه حل کارآمد استقرار خدمات اینترنت اشیا را بر اساس یک رویکرد مستقل برای استقرار برنامه های کاربردی اینترنت اشیا در زیرساخت مه پیشنهاد کردند. راه حل پیشنهادی کیفیت خدمات مورد نیاز خدمات اینترنت اشیا و قابلیت های گره های مه موجود را برای تعیین یک طرح استقرار خدمات کارآمد با استفاده از روش فراابتکاری الگوریتم بهینه سازی نهنگ (WOA) نظارت می کند. علاوه بر این، مکانیسم مبتنی بر تکامل نهنگ از پهنای باند و مصرف انرژی به عنوان توابع هدف برای یافتن طرح بهینه استقرار خدمات اینترنت اشیا استفاده می کند و در عین حال کیفیت خدمات مورد نیاز هر سرویس اینترنت اشیا را برآورده می کند. همچنین در این مطالعه، ساختار فضای خدماتی مستقلی را با توجه به معماری سه لایه اکوسیستم مه ایجاد می کنیم تا تعامل بین اجزای اصلی دستگاه اینترنت اشیا و لایه های مه برای میزبانی اینترنت تضمین شود. برنامه های کاربردی چیزها نتایج شبیه سازی نشان می دهد که راه حل پیشنهادی نسبت استفاده از منابع و نسبت پذیرش خدمات را افزایش می دهد و تأخیر سرویس و مصرف انرژی را در مقایسه با سایر مکانیسم های مبتنی بر فراابتکاری کاهش می دهد.
* اینترنت اشیا شامل زیرساخت برنامه های کاربردی و لایه های نرم افزاری بزرگی است که توسعه برنامه کاربردی بصری و شفاف را امکان پذیر می کند. این محیط بسیار پراکنده و پرانرژی باید کیفیت خدمات مستقر شده را با در نظر گرفتن ناهمگونی قابلیت ها و پروتکل ها و همچنین تحرک کاربران و اشیا تضمین کند. زیرساخت استقرار برای ارائه ویژگی های مورد نیاز، از جمله پارادایم هایی مانند شبکه های تعریف شده توسط نرم افزار و محاسبات مه، مجدداً پیکربندی شده است. هدف مقاله گروه تحقیقاتی همکارانش[34] بررسی استقرار سرویس های اینترنت اشیا در معماری Fog است. تیم تحقیقاتی دجامی و همکارانش یک زیرساخت و مدل کاربردی IoT و همچنین یک استراتژی قرار دادن پیشنهاد کردند که مصرف انرژی سیستم و به حداقل رساندن نقض برنامه ها را با استفاده از الگوریتم بهینه سازی ازدحام ذرات گسسته (DPSO) در نظر می گیرد. شبیه سازی ها با شبیه ساز iFogSim انجام شد و نتایج نشان دهنده اثربخشی روش پیشنهادی در مقایسه با رویکردهای مقایسه شده است.
* همان طور که ارتباطات برای سازگاری با برنامه های کاربردی جدید مانند واقعیت افزوده و واقعیت مجازی تکامل می یابد، پارادایم های نوظهور جدید ویژگی های مهمی مانند سطوح تأخیر کم، پشتیبانی از تحرک و آگاهی موقعیتی را ارائه می دهند. همان طور که در مورد محاسبات مه، که از پارادایم محاسبات ابری آشنا با آوردن قابلیت های پردازش، ارتباط و ذخیره سازی به لبه شبکه گسترش می یابد. ارائه این ویژگی های جدید همچنین چالش های جدیدی را ایجاد می کند که نیاز به طراحی و اجرای مکانیسم های ارکستراسیون برای مقابله با مدیریت منابع دارد. یکی از این مکانیسم ها مربوط به قرار دادن سرویس است که بر اساس اهداف بهینه سازی خاص، نقطه اجرای برنامه مناسب را انتخاب می کند. مطالعه ای توسط کریما و همکارانش[35]، یک مدل برنامه ریزی خطی عدد صحیح برای قرار دادن سرویس ها با هدف کاهش تأخیر برنامه های کاربردی محبوب پیشنهاد کرد. علاوه بر این، کاوشی بر اساس الگوریتم PageRank به نام Popularity Ranking Placement نیز معرفی شده است. نتایج شبیه سازی نشان می دهد که اکتشافی زمان اجرای کوتاه تری دارد و در حالی که نزدیک به سطح تأخیر راه حل مبتنی بر ILP است، می تواند بار روی گره های شبکه را بهتر متعادل کند.
* شبکه های اینترنت اشیا (IoT) که به خدمات ابری متکی هستند، معمولاً به دلیل عدم تضمین زمان پاسخگویی، از برنامه های بلادرنگ پشتیبانی نمی کنند. مدل محاسبه مه برای غلبه بر این مشکل با تکمیل وظایف در لبه شبکه که در آن محدودیت های زمانی قابل ارائه است، استفاده می شود. یک مسئله دشوار در معماری مبتنی بر مه، استقرار وظایف بر روی دستگاه های لبه ای برای دستیابی به عملکرد بهتر است. فرآیند نگاشت یک کار به یک دستگاه کامپیوتری به عنوان مشکل مکان سرویس (SPP) شناخته می شود. در مقاله ای توسط تیم تحقیقاتی تحقیقاتی خسروآبادی و همکارانش[36]، یک الگوریتم اکتشافی برای حل SPP به نام خوشه بندی دستگاه مه و خدمات حساس به تقاضا (SCATTER) ارائه شد. در این مطالعه، شبیه سازی هایی با استفاده از جعبه ابزار iFogSim و ارزیابی های تجربی با استفاده از سخت افزار واقعی برای تأیید امکان سنجی الگوریتم SCATTER با توجه به کاربردهای خانه هوشمند ارائه شد. روش SCATTER پیشنهادی با دو روش موجود مرتبط با اندازه گیری کیفیت خدمات (QoS) مقایسه می شود: یک رویکرد با ساختار لبه و یک رویکرد فقط ابری. نتایج تجربی نشان داد که روش SCATTER پیشنهادی به ترتیب با 42.1% و 60.2% زمان پاسخگویی، 45% و 65.7% تأخیر و 2.33% و 3.2% مصرف برق بهتر از روش مبتنی بر ساختار لبه و ابری عمل کرد.
* اینترنت اشیا یکی از توانمندسازهای کلیدی برای آگاهی موقعیتی مورد نیاز برای دستیابی به شهرهای هوشمند است. دستگاه های اینترنت اشیا، به ویژه برای اهداف نظارتی، الزامات زمان بندی سختی دارند که محاسبات ابری نمی تواند آنها را برآورده کند. این نقطه ضعف رایانش ابری را می توان با محاسبات مه که در آن گره های مه در نزدیکی دستگاه های اینترنت اشیا قرار می گیرند، برطرف کرد. از آنجایی که قابلیت های گره های مه در مقایسه با مراکز داده ابری کم است، گره های مه را نمی توان با تمام خدمات مورد نیاز دستگاه های اینترنت اشیا استفاده کرد. از این رو، گروه تحقیقاتی جونگ هاو و همکارانش[37] بر مشکل استفاده از خدمات مه متمرکز شدند و آخرین روند تحقیقاتی را در این زمینه ارائه کردند. بیشتر ادبیات مربوط به استقرار سرویس مه با تعیین نقطه مه مناسبی که الزامات مختلفی مانند تأخیر را از منظر یک یا چند درخواست سرویس برآورده می کند، مربوط می شود. هدف محقق در این تحقیق این است که به جای قرار دادن سرویس های درخواستی برای یک یا چند درخواست خدمات، خدمات مه شکن را با توجه به درخواست های خدماتی که ممکن است در دوره قبل جمع آوری شده است، به طور مؤثر قرار دهد. مفهوم شبکه مه منطقی اخیراً به دلیل مقیاس پذیری استفاده از خدمات مه در شهرهای هوشمند در مقیاس بزرگ مطرح شده است. یک شبکه مه منطقی در یک توپولوژی درختی که در مرکز داده ابری ریشه دارد تشکیل می شود. بر اساس شبکه مه منطقی، یک رویکرد قرار دادن سرویس پیشنهاد شده است به طوری که خدمات را می توان در گره های مه به شیوه ای کارآمد از منابع قرار داد.
* با افزایش تعداد دستگاه های اینترنت اشیا که منابع محدودی دارند، توسعه ای از پارادایم رایانش ابری به اصطلاح محاسبات مه پدیدار شده است که در آن تمام سلول های مه در لبه شبکه و زمان تأخیر قرار دارند. را می توان کاهش داد. در همین حال، چالش مهمی که با تعریف محاسبات مه توجه زیادی را به خود جلب کرده است، مسئله مکان یابی سرویس است که هنوز در ابتدای تحقیقات خود است. این اجازه می دهد تا برنامه های اینترنت اشیا را بر روی منابع مه محاسباتی، با هدف بهینه سازی کیفیت خدمات الزامات برنامه ها و در عین حال در نظر گرفتن حداکثر استفاده از منابع مه، استقرار دهید. گروه تحقیقاتی ایوبی و همکارانش[38]، یک روش مستقل استقرار سرویس اینترنت اشیا شامل چهار مرحله نظارت، تجزیه و تحلیل، تصمیم گیری و اجرا به نام (MADE) پیشنهاد کردند. ابتدا، منابع موجود و وضعیت خدمات برنامه در زمان اجرا نظارت می شود. در مرحله بعد، خدمات درخواستی با توجه به مهلت سرویس اپلیکیشن اولویت بندی می شوند. سپس، الگوریتم تکاملی پارتو برای تصمیم گیری در مورد جای گذاری سرویس برنامه به عنوان یک مسئله بهینه سازی چند هدفه اعمال می شود. در نهایت تصمیمات اتخاذ شده در مراحل قبلی در محیط مه اجرا می شود. نتایج آزمایش نشان می دهد که روش پیشنهادی از نظر معیارهای عملکرد متفاوت از همتایان خود بهتر عمل می کند.

در ادامه رویکردهای بررسی شده، به همراه مزایا و معایب موجود به طور کلی در یک جدول ارزیابی می شوند.

در ادامه تعدادی از رویکردهای ارائه شده در این زمینه به تفکیک مورد بررسی قرار خواهد گرفت.

# 1-3-2) زمان‌بندی کار عظیم انرژی و تأخیری در سیستم محاسباتی لبه و مه

در سیستم های محاسباتی لبه و مه، داده ها بسیار فشرده هستند و بر روی نزدیک ترین دستگاه های مه یا پلتفرم های ابری الاستیک و توزیع شده اجرا می شوند. مشکل در نظر گرفته شده شامل چندین دستگاه مه و چندین پلتفرم ابری است، که در آن هر کار زمان کوتاهی (چند میلی ثانیه) برای اجرا نیاز دارد. زمان اجرای یک کار در دستگاه مه یا پلتفرم ابری به تعداد کارهایی که در همان دستگاه در صف هستند بستگی دارد. وظایف در ابتدا توسط نزدیک ترین دستگاه های مه شکن جمع آوری می شوند و سپس برخی از آنها به دلیل ظرفیت محدود دستگاه های محاسباتی مه به پلتفرم های ابری مناسب تخلیه می شوند. از آنجایی که این وظایف داده فشرده هستند، تأخیرهای ارتباطی از مه به ابر قابل چشم پوشی نیست.

این مسائل زمان بندی کار در بسیاری از برنامه های محاسباتی مه-ابر بلادرنگ گسترده است. به عنوان مثال، در سیستم های تولید پیشرفته، خطوط مونتاژ خودکار با دستگاه های مهر مستقر می شوند که وضعیت دستگاه را مورد نظارت قرار می دهندو به طور دوره ای تشخیص عیب را بر اساس داده های تولید در زمان واقعی جمع آوری شده انجام می دهند. دستگاه های مه، داده های جمع آوری شده را (در یک واحد زمانی) به عنوان مجموعه ای از وظایف فشرده داده پردازش می کنند. این وظایف باید در یک واحد زمانی تکمیل شوند در غیر این صورت وظایف متوالی به تعویق می افتد. در یک کلام، نرخ سرویس (یا توان عملیاتی سیستم) باید رضایت بخش باشد. بنابراین، به وظایف جمع آوری شده در یک واحد زمانی، مهلتی اختصاص داده می شود که نمی تواند از طول یک واحد زمانی تجاوز کند. به طور کلی، دستگاه های مه در نزدیکی منابع داده قرار دارند، بنابراین تأخیرهای ارتباطی نسبت به تأخیر اجرای کار نسبتاً کمتر است. آنها می توانند در زیر بار کاری سطح پایین به سرعت پاسخ دهند. با این حال، آنها در توانایی محاسباتی محدود هستند. برخلاف دستگاه های مه، پلتفرم های ابری از منابع داده دور هستند و پهنای باند محدود شبکه ممکن است منجر به تأخیر بزرگ انتقال داده شود. از آنجایی که سناریوهای سیستم مه-ابر هر دو منابع مه و ابر را شامل می شود، تخلیه محاسباتی و زمان بندی وظایف نقش مهمی را برای کارایی و اثربخشی سیستم ایفا می کند.

در این راستا گروه تحقیقاتی جیا و همکارانش[8] مسئله بهینه سازی بارگذاری محاسباتی را با منابع ناهمگن در یک سیستم محاسباتی ابر-مه در نظر می گیرند. این مسئله در بسیاری از برنامه های بی درنگ و سیار رایج است، جایی که وظایف عظیم و محاسباتی فشرده هستند، و منابع محاسباتی می توانند هم دستگاه های مه و هم پلتفرم های ابری را شامل شوند. چالش ها در پیشنهاد الگوریتم های مؤثر، کارآمد و قوی با هدف به حداقل رساندن تأخیر کل و مصرف انرژی نهفته است. در این پژوهش یک مدل زمان بندی کار دوهدفه فرموله شده است، که در آن مدل های صف برای تخمین تأخیر و مدل های مصرف انرژی برای منابع ناهمگن معرفی می شوند. یک چارچوب زمان بندی وظایف عظیم مبتنی بر بهینه سازی پارتو برای زمان بندی وظایف عظیم در یک واحد زمانی پیشنهاد شده است. این رویکرد از یک مجموعه راه حل غیر نظارتی به دست آمده توسط انرژی ابزار و روش های جستجوی محلی آگاه از تأخیر انتقال شروع می شود. یک روش جستجوی محلی مبتنی بر درخت برای بهبود بیشتر راه حل های غیرمسلط پیشنهاد شده است.

همچنین در این پژوهش چارچوب زمان بندی وظایف عظیم مبتنی بر بهینه سازی پارتو برای مسئله مورد مطالعه پیشنهاد شده است که از دو جزء اصلی تشکیل شده است: روش تولید نامزد اولیه (ICGM) و فرایند اکتشافی تکراری مبتنی بر درخت. این فرایند راه حل های اولیه را توسط جستجوی محلی آگاه از انرژی و جستجوی محلی آگاه از تأخیر انتقال تولید می کند. جستجوی محلی آگاه از تأخیر، انتقال راه حل های جدید غیر نظارتی را در درخت جستجو با سطح محدود جستجو می کند. همچنین در این پژوهش، روش تولید همسایگی برای تولید راه حل های جدید در طول روند جستجوی محلی معرفی شده است. محققان عملیات افزایش متغیر را در تولید همسایگی برای تولید ماتریس های راه حل جدید بر اساس راه حل داده شده تعریف می کنند. الگوریتم پیشنهادی با چهار الگوریتم کلاسیک برای مسائل مشابه مقایسه شده است. عملکرد آنها با معیارهای بهینه سازی پارتو در جنبه های مختلف ارزیابی شده و نتایج نشان دهنده کارایی رویکرد پیشنهادی بود.

# 2-3-2) زمان‌بندی وظایف در سیستم‌های محاسباتی مه ابری

محاسبات مه، خدمات رایانش ابری را به کاربر نهایی نزدیک می کند و امکان پشتیبانی از برنامه های حساس به زمان را مانند برنامه های مربوط به خودروهای خودران، کنترل سیستم های روباتیک و بانکداری آنلاین فراهم می کند. این برنامه ها دارای مهلت های پردازشی مرتبط هستند و اعتبار خروجی آنها به زمانی که در دسترس می شوند بستگی دارد. اگر یک برنامه بلادرنگ نتواند ضرب الاجل خود را برآورده کند، عواقب آن می تواند از نتایج بی فایده تا رویدادهای فاجعه بار باشد. علاوه بر این، محاسبات مه همچنین از برنامه هایی که نیازمند تحرک، توزیع جغرافیایی، آگاهی از موقعیت مکانی و تجزیه و تحلیل زمان واقعی هستند، پشتیبانی می کند. برنامه های غیرواقعی می توانند روی گره های مه یا در ابر پردازش شوند. به عنوان مثال، ادغام داده های اینترنت اشیا (IoT) را می توان بر روی گره های مه با منابع محدود انجام داد، در حالی که برنامه های کاربردی در مقیاس بزرگ مانند مواردی در سیستم های مراقبت های بهداشتی و پخش ویدئو HD به دلیل نیاز به فضای ذخیره سازی و قدرت پردازش زیادی دارند، امکان استفاده از این فرایند را نخواهند داشت. به طور کلی چندین مسئله در شبکه های اینترنت اشیا و مه ابری در اصل مسائل بهینه سازی هستند و یکی از این موارد، مسئله زمانبندی است. هنگامی که یک برنامه کاربردی به لبه شبکه می رسد، وظایفی که این برنامه کاربردی را تشکیل می دهند باید بر روی عناصر پردازشی برنامه ریزی شوند. زمانبند مسئول نگاشت وظایف یک برنامه کاربردی بر روی منابع موجود است تا بتوان به اهداف خاصی دست یافت.

یک زمان بند، زمانی را مشخص می کند که در آن منابع باید توسط برنامه ها استفاده شوند و باید متغیرهای ورودی مرتبط با برنامه، مانند نیازهای منابع و الزامات کیفیت خدمات (QoS) و همچنین متغیرهای مرتبط با در دسترس بودن منابع با توجه به تنوع الزامات QoS مورد نیاز برنامه های کاربردی در زنجیره اجرایی در دسترس باشد. چالش اصلی در چنین محیط ناهمگون، استفاده از تکنیک های زمان بندی مؤثر برای تضمین پایان برنامه ها در بازه زمانی مورد نیاز است. در سناریویی که در پژوهش گروه تحقیقاتی جودی و همکارانش[39] در نظر گرفته شده است، کاربران نهایی برنامه هایی را برای پردازش در یک سیستم ابری-مه ارسال می کنند. یک طبقه بندی کننده در لبه شبکه، برچسب هایی را به برنامه ها اختصاص می دهد تا کلاس خدمات (CoS) آن ها را با توجه به نیازهای QoS شناسایی کنند. پس از آن، برچسب و گردش کار برنامه به یک زمان بند ارسال می شود، که هر وظیفه برنامه را بر روی گره های ابر / مه با منابع موجود برای برآوردن نیازهای برنامه ترسیم می کند.

این مقاله الگوریتم های زمان بندی کار را برای پشتیبانی از خدمات چند کلاسه در سیستم های محاسباتی مه ابری معرفی می کند. هیچ یک از پیشنهادات قبلی برای زمان بندی وظایف در سیستم های لبه و مه، برنامه ها را برای تصمیم گیری در مورد اینکه وظایف باید در کجا پردازش شوند، در نظر نگرفتند. علاوه بر این، الگوریتم های زمان بندی پیشنهادی انواع مختلفی از منابع مانند CPU، RAM و فضای ذخیره سازی را در نظر می گیرند. زمانبندی پیشنهادی بر اساس فرمول برنامه ریزی خطی، دو راهکار ارائه کرده است. اولی که CASSIA-INT نام دارد، یک فرمول برنامه نویسی عدد صحیح دقیق را حل می کند، در حالی که دومی به نام CASSIA-RR، یک نسخه از فرمول برنامه نویسی عدد صحیح را حل می کند. CASSIA-RR نتایج تقریبی نزدیک به نتایج ارائه شده توسط CASSIA-INT را در مدت زمان بسیار کوتاه تری تولید می کند و استفاده از آن برای سیستم های ابری-مه آلود که در آن ها باید تصمیمات سریع گرفته شود توصیه می شود.

یک الگوریتم تقریبی، CASSIA-RR، برای به حداقل رساندن طول برنامه های محاسباتی مه استفاده می شود. هر دو روند پیشنهادی به دنبال اجرای فرایندی با کمترین زمان هستند. اثربخشی و کارایی این سناریوها در ارزیابی های مختلف، با تعداد گره های پردازشی و سطوح استفاده متفاوت نشان داده شد. الگوریتم های CASSIA-INT و CASSIA-RR از الگوریتم های زمان بندی سنتی مانند Random و Round Robin از نظر طول ساخت بهتر عمل می کنند. الگوریتم های CASSIA-INT و CASSIA-RR را می توان به راحتی در مدیریت منابع یک سیستم ابری ادغام کرد. آنها با پیشنهادات قبلی با در نظر گرفتن کیفیت سرویس برنامه برای تصمیم گیری در مورد اینکه وظایف یک برنامه باید در کجا پردازش شوند، متفاوت هستند. نتایج به دست آمده نشان می دهد که برای همه سناریوها، مقادیر makepan به دست آمده از هر دو زمان بندی کوچک تر از سایر رویکردها است. علاوه بر این، زمان مورد نیاز برای CASSIA-RR برای تولید یک زمان بندی به طور قابل توجهی کمتر از زمان بندی مورد نیاز CASSIA-INT است، که برای زمان بندی کننده های محیط های بلادرنگ بسیار مطلوب است.

# 3-3-2) زمان‌بندی وظایف چند هدفه در محاسبات ابری و مه با استفاده از رویکرد برنامه نویسی هدف

یکی از کاربردهای رایانش ابری در اینترنت اشیا جمع آوری و تجزیه و تحلیل داده های اینترنت اشیا است. مرکز ابر با دریافت داده های مختلف از انواع دستگاه های اینترنت اشیا، نتایج مربوطه را تجزیه و تحلیل کرده و سیگنال های کنترلی لازم را به این اشیا ارسال می کند. در این فرایند، چالش ها در سمت سرورهای ابری و همچنین در سمت اینترنت اشیا رخ می دهد. رویکرد محاسبات مه نقش واسطه ای را در اتصال دستگاه های اینترنت اشیا به مرکز ابر ایفا می کند تا این مشکلات و چالش ها را حل کند. دستگاه های اینترنت اشیا در یک شبکه محلی به گره های مه متصل می شوند و تمام اتصالات آنها به مرکز ابر از طریق گره های مه برقرار می شود. گره های مه می توانند تمام اطلاعات نزدیک اینترنت اشیا را دریافت کنند و سپس فقط اطلاعات قابل استفاده در مورد دستگاه های اینترنت اشیا را به ابر ارسال کنند و از این طریق ترافیک شبکه را کاهش داده و عملکرد کلی شبکه را افزایش دهند. از سوی دیگر، دستگاه های اینترنت اشیا اغلب قدرت پردازش ضعیفی دارند و می توانند از قدرت پردازش گره های مه استفاده کنند، بنابراین موضوع زمان بندی وظایف بین گره های مه مطرح شده است و برخی از محققان بر روی زمان بندی وظایف در محاسبات مه کار کرده اند. در اینجا، به دلیل ناهمگونی گره های مه و ابر و همچنین تأخیرهای شبکه بین اینترنت اشیا و این سرورهای پردازشی، بهینه سازی زمان بندی وظایف، چالشی برای محققان است.

در این راستا، برخی از محققین بر تعادل بار و مصرف بهینه انرژی در سرورها برخی نیز بر یافتن بهترین راه حل برای کاهش زمان ساخت و یا تأخیر سرویس تمرکز کرده اند. همچنین در برخی از مقالات، علاوه بر زمان، به هزینه و مبادله بین آنها پرداخته شده است. برای سادگی، اکثر الگوریتم ها این مسئله را با استفاده از روش جمع وزنی که یک راه ساده برای حل یک مسئله چند هدفه است، به یک مسئله تک هدفه تبدیل کرده اند. یکی از روش های مناسب برای حل مسئله چندهدفه، رویکرد برنامه ریزی هدف (GPA) است که در مقاله گروه تحقیقاتی نجف زاده و همکارانش[40] به حل مسئله با این روش پرداخته شده است. از سوی دیگر، محققانی که به موضوع زمان بندی وظایف پرداخته اند، به محدودیت اجرای کار بر روی گره های مه و ابر توجهی نکرده اند. مشتری اینترنت اشیا ممکن است بخواهد وظیفه خود را به روشی کنترل شده بر روی گره های مه محلی انجام دهد.

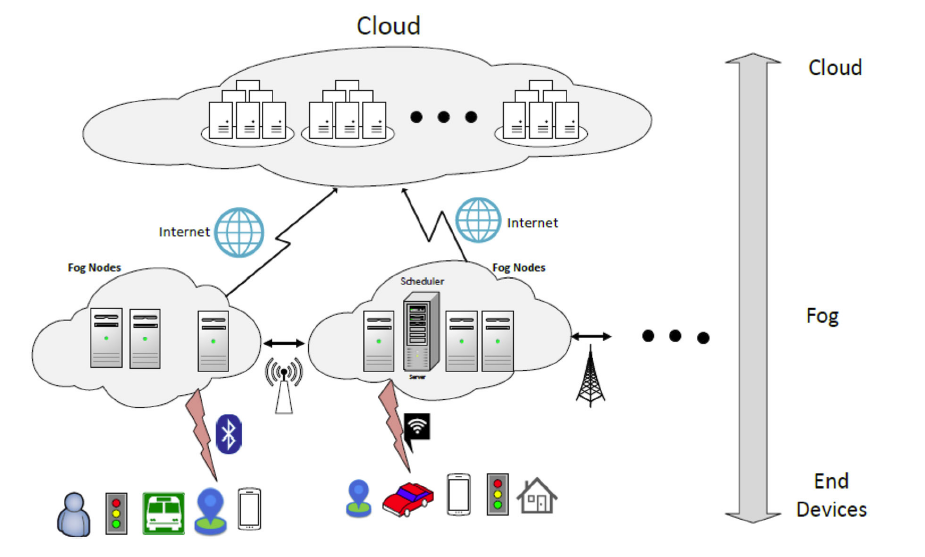
در این صورت باید مکانیزمی برای رعایت این شرایط در موضوع در نظر گرفته شود که در این نوشتار به آن خواهیم پرداخت. سهم اصلی این تحقیق به شرح زیر است.

* در این پژوهش یک پارامتر جدید به نام سطح دسترسی پیشنهاد می شود که درخواست کننده وظیفه (دستگاه اینترنت اشیا) می تواند مشخص کند که وظیفه می تواند توسط هر گره یا فقط توسط گره های مه محلی اجرا شود.
* محققان در این پژوهش یک الگوریتم زمانبندی کار چند هدفه در Cloud-Fog بر اساس الگوریتم بازپخت شبیه سازی شده پیشنهاد می کنند که چهار هدف مهم زمان بندی شامل زمان اجرای سرویس، هزینه اجرای سرویس، مهلت و سطح دسترسی را در نظر می گیرد.
* برای حل اختلاف بین زمان سرویس و هزینه خدمات، از GPA استفاده شده است. همچنین با ارائه یک مدل ریاضی کامل، دو هدف مهلت و سطح را به عنوان شرایط در مسئله زمان بندی تدوین شده است.

همان طور که در شکل 3-1 نشان داده شده است، معماری سیستم در سه لایه ایجاد شده است که شامل گره های ابری، گره های مه و دستگاه های پایانی هستند. پروتکل ارتباطی بین لایه سوم (IoT) و لایه دوم (محاسبات مه) می تواند انواع مختلفی مانند Wi-Fi، Bluetooth، Bluetooth Low Energy (BLE)، ZigBee، ANT، Z-Wave، 6LowPAN، Thread، WiMax، Cellular، LPWAN، RFID و NFC و پروتکل بین لایه دوم و لایه اول (رایانش ابری) مبتنی بر IP بوده و سرعت بالاتری دارد. در شکل 3-1 اجزای سطح پایین تر از اجزای بالا قدرت پردازش کمتری دارند، در حالی که اجزای بالایی هزینه بیشتری نسبت به اجزای پایین تر دارند. کنترل دسترسی گره های ابری برای اجرای درخواست های کاربر در ابتدا توسط اینترنت اشیا تعیین می شود، به طوری که دستگاه اینترنت اشیا متقاضی، همراه با درخواست خود، مشخص می کند که سطح دسترسی به وظیفه او در مورد «نحوه اجرای وظیفه» یکی از سطوح زیر:

* **دسترسی آزاد:** هر دو گره ابر و گره مه می توانند کار را اجرا کنند.
* **دسترسی محدود:** فقط گره های مه محلی توانایی اجرای وظیفه کاربر را دارند.

داده ها در هر دو سطح در طول انتقال از طریق شبکه رمزگذاری می شوند. در سطح دسترسی دوم، گره مدیر برنامه ریز در صورتی که منابع کافی برای اجرا بر روی گره های مه محلی را نداشته باشد، آن درخواست را رد می کند. در این صورت کاربر می تواند با افزایش محدودیت مهلت از رد کار جلوگیری کند. درخواست های دستگاه های IoT ابتدا به گره مه محلی ارسال می شود. سپس با توجه به پیکربندی سیستم، این گره مه محلی اطلاعات این درخواست را برای گره مدیر (مسئول زمان بندی وظایف) که در یکی از خوشه های مه قرار دارد ارسال می کند. در شکل 3-1، گره مدیر در دومین خوشه مه قرار دارد. پس از اجرای برنامه زمانبندی کار، گره مدیر وظیفه مربوطه را برای اجرا در یکی از گره های مه یا گره های ابری ارسال می کند.



**شکل 2- 2) مدل پیشنهادی گروه تحقیقاتی نجف زاده و همکارانش**[40]

با توجه به ماهیت چند هدفه الگوریتم پیشنهادی، محققان از GPA برای شناسایی مناسب ترین راه حل در بین راه حل های غیر غالب استفاده کردند. برای ارزیابی کارایی الگوریتم، الگوریتم پیشنهادی با سایر الگوریتم های MOPSO، MOTS و MOMF در چهار سناریو مختلف (آسان، متوسط، نیمه سخت و سخت) با روش مقایسه منصفانه (100 بار اجرا) مقایسه شد.. شبیه سازی ها نشان داد که الگوریتم MOSA پیشنهادی دو مزیت دارد، بهترین مبادله بین اهداف زمان و هزینه و همچنین کمترین تخطی از شرایط مربوط به دو هدف دیگر مسئله (کنترل سطح دسترسی و مهلت انجام وظایف). هنوز ایده ها و مسائل بالقوه ای در مورد زمان بندی وظایف در ابر-مه وجود دارد. از جمله در نظر گرفتن حریم خصوصی کاربر، در نظر گرفتن زمان بندی گردش کار، توسعه الگوریتم های زمان بندی با پارامترهای پویا، و همچنین ایجاد الگوریتم های فراابتکاری خلاقانه مناسب برای مسائل گسسته را می توان در تحقیقات آینده گنجاند.

# 4-3-2) مشخص کردن زمان‌بندی برنامه ها در منابع محاسباتی لبه، مه و ابر

رایانش ابری به یک سرویس محاسباتی توزیع شده محبوب تبدیل شده است که توسط ارائه دهندگان تجاری ارائه می شود. اخیراً، منابع محاسباتی لبه و مه در یک شبکه گسترده به عنوان بخشی از اینترنت اشیا ظهور کرده اند. این دو لایه انتزاع منابع مکمل یکدیگر هستند و مزایای متمایزی را ارائه می دهند. زمان بندی برنامه های کاربردی در ابرها یک حوزه فعال تحقیقاتی بوده است و با استفاده از مدل های گردش کار و جریان داده، یک ساختار انتزاعی انعطاف پذیر برای مشخص کردن برنامه های کاربردی ارائه می دهند. با این حال، مدل های برنامه نویسی و برنامه ریزی کاربردی برای لبه و مه هنوز در حال بلوغ هستند. در این راستا، گروه تحقیقاتی واراشنکی و همکارانش، یک طبقه بندی از مفاهیم ضروری برای مشخص کردن و حل مسئله زمان بندی برنامه ها در منابع محاسباتی لبه، مه و ابر را ارائه کردند. در این پژوهش ابتدا قابلیت ها و محدودیت های منابع این زیرساخت ها مشخص شدند و طبقه بندی مدل های کاربردی، محدودیت ها و اهداف کیفیت خدمات و تکنیک های زمان بندی بر اساس پژوهش های موجود ارائه شدند همچنین نمونه های اولیه تحقیقاتی و مقالات کلیدی با استفاده از این طبقه بندی بررسی شدند. این پژوهش به توسعه دهندگان و محققین این منابع توزیع شده در طراحی و دسته بندی برنامه های کاربردی، انتخاب انتزاع محاسباتی مربوطه، و توسعه یا انتخاب الگوریتم زمان بندی مناسب، سود می رساند. همچنین شکاف هایی را در ادبیات که مشکلات باز باقی می مانند برجسته می کند.

# 5-3-2) زمان‌بندی کار آگاه از هزینه در محیط مه-ابر

برای برنامه های حساس به تأخیر، پاسخ های دیرهنگام ممکن است منجر به خطا در سیستم یا حتی خرابی سیستم شود. به عنوان مثال، در یک برنامه کاربردی سلامت هوشمند که وضعیت بیمار را کنترل می کند، پاسخ های دیرهنگام ممکن است زندگی انسان را به خطر بیندازد. وقتی حجم زیادی از داده های تولید شده توسط دستگاه های اینترنت اشیا را در نظر بگیریم، این امر حتی مهم تر می شود. به عنوان مثال، در برنامه های شهر هوشمند، دستگاه های IoT حدود یک میلیون رکورد در ثانیه تولید می کنند. با توجه به تراکم بالای سنسورها در تعدادی از کاربردها و دقت کم اندازه گیری، امکان تکرار و خطاپذیری داده ها در محیط های IoT قابل توجه است بنابراین، این داده های تکراری و مستعد خطا، ممکن است اندازه داده های غیر ضروری را افزایش دهند. انتقال حجم عظیمی از داده ها و درخواست ها به ابر ممکن است منجر به استفاده کم از منابع شبکه، تأخیر زیاد در انتقال، هزینه های مالی و سربار پردازش و ازدحام شبکه شود. به گفته محققان، محاسبات مه به عنوان یک راه حل امیدوارکننده، یک پلتفرم مجازی است که ظرفیت های پردازش، ذخیره سازی و شبکه را بین دستگاه های IoT و منابع رایانش ابری فراهم می کند و به طور کلی، اما نه منحصراً، مه در لبه شبکه قرار دارد.

یک معماری رایج محاسبات مه شامل سه لایه اصلی است: دستگاه، مه و لایه ابر. محاسبات مه یک لایه میانی بین لایه های دستگاه و ابر فراهم می کند که شامل تعداد زیادی گره مه است که قابلیت پردازش، شبکه سازی و غیره را دارند. از آنجایی که قرار است گره های مه در نزدیکی لایه دستگاه قرار گیرند، درخواست های دارای الزامات تأخیر بالا را می توان در لایه مه پردازش کرد و سایر درخواست ها را می توان به ابر ارسال کرد. علاوه بر این، از آنجایی که قابلیت های پردازش محلی را فراهم می کند، می تواند استفاده از پهنای باند و هزینه مالی را کاهش دهد. لایه مه، درخواست ها و داده های دریافتی را فیلتر، تجزیه و تحلیل و پیش پردازش می کند. اگر لایه مه منابع کافی در دسترس داشته باشد و بتواند وظیفه ای را اجرا کند، کار در لایه مه برنامه ریزی می شود. در غیر این صورت، اگر منابع کافی برای اجرای کار وجود نداشته باشد، به لایه ابری ارسال می شود. اگر هیچ یک از آنها (یعنی لبه و مه) نتوانند در زمان مشخص شده پاسخ دهند، کار رد می شود. بنابراین، یافتن مکان بهینه خدمات و منابع در معماری اینترنت اشیا سه تایر برای دستیابی به بهبود هزینه و کارایی منابع، QoS بالاتر و سطح بالاتری از حریم خصوصی امنیتی مهم است. علاوه بر این، لایه مه ممکن است به طور همگن یا ناهمگن سازماندهی شود.

در ساختار همگن، منابع مه دارای قابلیت های پردازش، ذخیره سازی و پهنای باند یکسانی هستند، اما در حالت ناهمگن، منابع دارای قابلیت های متفاوتی هستند. در مورد دستگاه های مه ناهمگن، زمان بندی وظایف از پیچیدگی بیشتری برخوردار است، زیرا منابع دارای قابلیت های متفاوتی از نظر سرعت پردازش، جابه جایی و غیره هستند که منجر به ترکیب های مختلفی (وظیفه و منبع) می شود. جدا از زمان پاسخگویی و ضرب الاجل برای هر کار، پارامترهای دیگری مانند بهره وری منابع، انرژی و هزینه مالی نیز ممکن است لازم باشد توسط الگوریتم های زمان بندی کار در نظر گرفته شوند. در نظر گرفتن چندین پارامتر در کنار ناهمگونی وظایف و منابع، پیچیدگی بیشتری را بر فرآیند تصمیم گیری تحمیل می کند. با افزایش تعداد کارها و منابع، راه حل های ممکن به صورت تصاعدی رشد می کنند و هیچ الگوریتم چند جمله ای قطعی برای حل این مسئله وجود ندارد و به عنوان یک مسئله NP-Hard در نظر گرفته می شود. برای حل این نوع مسائل می توان از الگوریتم های تکاملی مانند بهینه سازی ازدحام ذرات (PSO)، الگوریتم ژنتیک (GA)، الگوریتم زندگی زنبورها (BLA) و بهینه سازی کلونی مورچه ها (ACO) استفاده کرد. با توجه به بهینه سازی جهانی و توانایی جستجوی GA، به طور گسترده ای برای حل مسائل زمان بندی کار استفاده می شود.

در این راستا، گروه تحقیقاتی نیکویی و همکارانش[16]، زمان بندی کار را در یک سیستم محاسباتی مه ابری در نظر می گیرند، جایی که گره های مه با گره های ابری اجاره ای برای اجرای مؤثر برنامه های بارگذاری در مقیاس بزرگ کاربران همکاری می کنند. الگوریتم پیشنهادی تصمیم می گیرد که آیا وظیفه ای را در لایه مه اجرا کند یا به لایه ابری ارسال کند. مشارکت های اصلی این کار به شرح زیر است:

* کاهش هزینه اجرای وظایف در لایه مه / ابر، که شامل هزینه پهنای باند مورد استفاده (برای ارسال / دریافت درخواست ها و پاسخ ها) و هزینه استفاده از منابع محاسباتی است.
* افزایش میزان موفقیت در پاسخگویی به وظایف در مهلت مقرر

ساختار مدل پیشنهادی این گروه تحقیقاتی در شکل زیر نشان داده شده است.



**شکل 2- 3) مدل پیشنهادی گروه تحقیقاتی نیکویی و همکارانش**[16]

در این پژوهش زمان بندی کار در یک سیستم محاسباتی مه ابری در نظر گرفته می شود، که در آن گره های مه با گره های ابری اجاره ای برای اجرای مؤثر برنامه های بارگذاری در مقیاس بزرگ کاربران همکاری می کنند. این مقاله یک الگوریتم مبتنی بر ژنتیک را پیشنهاد می کند که CAG نامیده می شود، با در نظر گرفتن مبادله بین توان عملیاتی و هزینه برای کاربردهای سخت زمان واقعی. برای ارزیابی عملکرد رویکرد پیشنهادی، در این پژوهش تغییراتی در شبیه ساز iFogSim انجام شد و CAG در موارد مختلف برای بررسی تأثیر تغییرات بر عملکرد ارزیابی شد. نتایج نشان می دهد که CAG از نظر هزینه مالی و میزان موفقیت، بهتر از الگوریتم های RR و حداقل زمان پاسخ گویی عمل می کند.

# 4-2) مقایسه روش‌های موجود

در این فصل از پژوهش، رویکردهایی در زمینه بهبود فرایند زمانبندی در محیط محاسباتی مه و لبه مورد بررسی قرار گرفتند و نقاط قوت و ضعف آنها ارائه شد. در ادامه و در جدول (2-1) مقایسه کلی از رویکردهای بررسی شده ارائه می شود و با درنظر گرفتن نتایج مقایسه، پیشنهاد کار آتی در فصل بعدی بیان می شود.

جدول 2- 1) جدول مقایسه روش‌های موجود

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **مؤلف** | **روش پیشنهادی** | **سرعت** | **تأخیر** | **مصرف انرژی** | **کیفیت سرویس** | **طول عمر شبکه** |
| گروه تحقیقاتی جیان و همکارانش[28] | ارائه روش کاملاً جدید برای مسئله جای‌گذاری سرویس که می‌تواند زمان قرارگیری ماشین مجازی و مصرف انرژی را کاهش دهد. | متوسط | متوسط | بالا | بالا | پایین |
| گروه تحقیقاتی الشماری و همکارانش[7] | تحلیلی از آخرین تکنیک‌های زمان‌بندی منابع برای محیط‌های یکپارچه اینترنت اشیا و محاسبات مه ارائه کردند | پایین | پایین | بالا | بالا | پایین |
| گروه تحقیقاتی غفاری و همکارانش[10] | نسخه جدیدی از بهینه‌سازی خرگوش‌های مصنوعی به نام بهینه‌سازی خرگوش‌های مصنوعی آشوب‌زده مبتنی بر غیرخطی را پیشنهاد می‌کند و از بهینه‌سازی خرگوش برای زمان‌بندی کار در محیط محاسبات مه استفاده می‌کند | پایین | متوسط | بالا | پایین | متوسط |
| **وانگ و همکارانش[2]** | یک الگوریتم زمانبندی مبتنی بر یادگیری تقویتی را پیشنهاد کردند که تا زمان پاسخ‌دهی برنامه های ناهمگن اینترنت اشیا را به‌طور سازگار و کارآمد بهینه کند و بار سرورهای لبه / مه را متعادل کند. | بالا | پایین | متوسط | پایین | متوسط |
| **هوسین و همکارانش[21]** | یک تکنیک زمان‌بندی کار به نام زمان‌بندی وظایف اولویت‌دار آگاه از منابع را در یک محیط محاسباتی مه ناهمگن پیشنهاد کردند. | متوسط | بالا | پایین | متوسط | متوسط |
| گروه تحقیقاتی صراف زاده و همکارانش[29] | به حداقل رساندن تأخیر برنامه و استفاده از شبکه با پیشنهاد یک الگوریتم قرار دادن سرویس مبتنی بر ژنتیک در محیط‌های ابر-مه | متوسط | پایین | بالا | بالا | پایین |
| گروه تحقیقاتی تقی زاده و همکارانش[30] | برای بهبود در دسترس بودن داده ها و کاهش تأخیر و هزینه دسترسی به داده، از فراابتکاری NSGA-II برای تخصیص داده ها در بهترین گره مه نزدیک کاربر استفاده کردند. | بالا | بالا | پایین | بالا | متوسط |
| گروه تحقیقاتی آرانی و همکارانش[31] | ارائه یک راه حل کارآمد تخصیص خدمات اینترنت اشیا با الهام از استفاده از الگوی محاسباتی خودمختار برای استقرار برنامه های اینترنت اشیا در زیرساخت مه مجازی | بالا | پایین | متوسط | متوسط | بالا |
| گروه تحقیقاتی سالیمان و همکارانش[32] | ارائه یک چارچوب محاسباتی مفهومی مستقل مبتنی بر میان‌افزار کنترل مه-ابر برای مدیریت بهینه مه و منابع ابر | متوسط | پایین | پایین | بالا | متوسط |
| گروه تحقیقاتی تکیه بند و همکاران [33] | بررسی مشکل ارائه خدمات پویا را برای ارائه خدمات کاربردی IoT بر روی زیرساخت مه ناهمگن و پویا | بالا | پایین | متوسط | بالا | بالا |
| گروه تحقیقاتی آرانی و همکارانش [31] | ارائه یک راه‌حل کارآمد قرار دادن سرویس اینترنت اشیا را بر اساس روش‌شناسی خودمختار برای استقرار برنامه های اینترنت اشیا در زیرساخت مه | متوسط | بالا | پایین | متوسط | متوسط |
| گروه تحقیقاتی دجامی و همکارانش[34] | بررسی قرارگیری خدمات اینترنت اشیا در معماری Fog | متوسط | متوسط | بالا | بالا | پایین |
| گروه تحقیقاتی خسروآبادی و همکارانش[36] | پیشنهاد یک الگوریتم اکتشافی را برای حل SPP که به‌صورت خوشه‌بندی دستگاه های مه و خدمات حساس به نیاز ابتدا (SCATTER) نام‌گذاری شده‌است. | بالا | بالا | پایین | بالا | متوسط |
| گروه تحقیقاتی جونگ هاو و همکارانش[37] | ارائه موضوع جای‌گذاری سرویس مه متمرکز شده و روندهای تحقیقاتی اخیر در این موضوع | بالا | پایین | متوسط | متوسط | بالا |
| گروه تحقیقاتی ایوبی و همکارانش[38] | ارائه یک روش مستقل استقرار سرویس اینترنت اشیا شامل چهار مرحله نظارت، تجزیه و تحلیل، تصمیم گیری و اجرا | متوسط | پایین | پایین | بالا | متوسط |
| گروه تحقیقاتی کریما و همکارانش[35] | پیشنهاد یک مدل برنامه‌ریزی خطی عدد صحیح برای قرار دادن سرویس با هدف کاهش تأخیر برنامه های کاربردی محبوب | بالا | پایین | متوسط | بالا | بالا |
| گروه تحقیقاتی هالوا و همکارانش[1] | جای‌گذاری سرویس بلادرنگ در چارچوب محاسباتی مه برای کاربردهای اینترنت اشیا: رویکرد مبتنی بر منطق فازی | پایین | پایین | پایین | بالا | بالا |
| گروه تحقیقاتی حسینی و همکارانش[22] | زمان بندی وظایف برای محاسبات مه داوطلبانه به نام‌های Min-CCV و Min-V | پایین | متوسط | متوسط | متوسط | متوسط |
| گروه تحقیقاتی صراف زاده و همکارانش[29] | به حداقل رساندن تأخیر برنامه و استفاده از شبکه با پیشنهاد یک الگوریتم قرار دادن سرویس مبتنی بر ژنتیک در محیط‌های مه | متوسط | بالا | متوسط | متوسط | بالا |
| گروه تحقیقاتی رینو و همکارانش[3] | ارائه الگوریتم جای‌گذاری سرویس جدید در محاسبات مه به نام برای بهینه‌سازی تابع چند هدفه | متوسط | بالا | متوسط | متوسط | بالا |
| گروه تحقیقاتی لاخان و همکارانش[23] | ارائه یک چارچوب الگوریتم جای‌گذاری سرویس کارآمد با در نظر گرفتن پارامترهای انرژی، بر اساس شبکه‌های عصبی عمیق | بالا | متوسط | متوسط | بالا | متوسط |
| گروه تحقیقاتی لیجز و همکارانش[4] | ارائه یک رویکرد جای‌گذاری سرویس را در محیط‌های مه- با در نظر گرفتن مسئله مصرف انرژی در زمان تکمیل برنامه | بالا | متوسط | بالا | متوسط | بالا |
| گروه تحقیقاتی معماری و همکارانش [11] | یک الگوریتم جای‌گذاری سرویس آگاه از تأخیر مبتنی بر تطبیق ماشین مجازی با استفاده از رویکردهای فراابتکاری | بالا | متوسط | بالا | متوسط | بالا |
| گروه تحقیقاتی رفعت و همکارانش[12] | ارائه یک راه‌حل جای‌گذاری سرویس به‌منظور برآورده کردن حداکثر تعداد درخواست‌ها با توجه به الزامات مهلت زمانی آنها | بالا | بالا | متوسط | بالا | متوسط |

# 5-2) خلاصه و نتیجه گیری

در این فصل از پژوهش، یک مرور کلی از رویکردهای زمابندی در معماری لبه و مه به همراه مزایا و معایب هر یک از رویکردهای بررسی شده ارائه شد. با توجه به مطالب بیان‌شده، در فصل آتی ضمن ارائه جمع بندی کلی، پیشنهادات آتی در این زمینه ارائه می شود.

3) فصل سوم

**راهکار پیشنهادی**

**اهداف فصل:**

* توصیف رهیافت و روش پژوهش
* ارائه طرح کلی و مدل مفهومی پژوهش
* ارائه مدل پیشنهادی پژوهش

# 1-3) مقدمه

امروزه محققان برای حل مسئله زمانبندی در محاسبات ابری و ساختارهای مه، از رویکردهای متنوعی استفاده کرده اند.

در تعدادی از رویکردها، از ساختارهایی با عنوان یادگیری ماشین و یادگیری عمیق به منظور حل مسئله زمانبندی در محیط لبه و مه استفاده کرده اند، در یادگیری عمیق، عامل به طور مداوم با محیط در تعامل است و تعداد زیادی از مسیرهای تجربی (به عنوان مثال، توالی حالت ها، اقدامات و پاداش ها) را ثبت می کند که در مرحله آموزش برای یادگیری سیاست های بهینه استفاده می شود. در این پژوهش نیز، یک رویکرد زمانبندی کاربردی اینترنت اشیا مبتنی بر یادگیری عمیق پیشنهاد شد که از ساختار شبکه عصبی حافظه طولانی کوتاه مدت به منظور طبقه بندی منابع پردازشی و اجرای فرایند زمانبندی استفاده می کند. الگوریتم پیشنهادی، از تکنیک طبقه بندی مبتنی بر یادگیری عمیق برای حل چالش زمانبندی برنامه های اینترنت اشیا در محیط های محاسباتی لبه و مه استفاده می کند. روش پیشنهادی می تواند به طور موثر هزینه متعادل سازی بار سرورها، هزینه زمان پاسخ برنامه های اینترنت اشیا و هزینه وزنی آنها را بهینه کند. علاوه بر این، با استفاده از روش پیشنهادی برای محدود کردن دفعات به روز رسانی پارامترهای کیفیت سرویس در هر تکرار، انتظار می رود سرعت همگرایی الگوریتم بهبود یابد.

# 2-3) روش شناسی پژوهش

* **روش تحقیق بر اساس نتایج:** از نظر حل مسئله و کاربرد در شبکه های مه و لبه استراتژیک و کاربردی و از نظر نوآوری در روش ها توسعه ای است. در نتیجه، با توجه به کسب و درک دانش جدید حوزه علمی، این تحقیق یک توسعه کاربردی است.
* **طبقه بندی پژوهش بر اساس فرآیند اجرا:** با توجه به ماهیت داده ها، این تحقیق کمی و مبتنی بر رویکرد کاربردی، تحلیلی- توصیفی است.
* **طبقه بندی تحقیقات بر اساس مهارت های اجرا:** با توجه به رویکرد کمی، پژوهش از نظر ماهیت استقرایی است.
* **طبقه بندی تحقیق بر اساس اهداف:** با بهره گیری از نتایج و در نظر گرفتن بهترین، اهداف اصلی و عملی، منجر به تدوین فرضیه و طراحی یک رویکرد زمانبندی پویا در ساختار محاسبات مه و لبه می شود، نوع تحقیق پژوهشی – کاربردی است.

# 3-3) مدل مفهومی پژوهش

پژوهش پیشنهادی در پنج مرحله اجرا می شود. در فاز اول، بیان مسئله، مفاهیم محاسبات ابری، اینترنت اشیا و ساختار مه و لبه در آن مورد بررسی قرار گرفته در ادامه مسئله زمانبندی در این ساختارها به عنوان چالش مطالعه می شود. در بخش دوم از پژوهش، به بررسی و ارزیابی مفاهیم مربوط به زمانبندی در محیط مه و لبه، الگوها و راهکارهای ارائه شده توسط محققان در این زمینه پرداخته می شود و در ادامه مزایا و معایب هریک از رویکردهای بررسی شده توصف می شود. در فصل سوم، روش پیشنهادی توصیف شده و ساختار آن به صورت کلی مورد ارزیابی قرار می­گیرد. در در فصل چهارم، شبیه سازی و ارزیابی روش پیشنهادی اجرا شده و نتایج به دست آمده از شبیه سازی، با رویکردهای موجود در این زمینه مقایسه می شود. و در نهایت در فصل پنجم جمع بندی رویکرد درنظر گرفته انجام و پیشنهاداتی برای ادامه و بهبود تحقیق و فرایند تخصیص منابع و زمانبندی جهت آیندگان ارائه می شود.

همان طور که بیان شد، مسئله زمانبندی در محاسبات مه و لبه یک مسئله غیرقطعی است که همواره روش های مختلفی برای حل آن ارائه می شود. یکی از کاراترین روش ها برای حل مسائل غیرقطعی استفاده از الگوریتم های تکاملی است. اما مسئله مدیریت اطلاعات در ساختارهای تکاملی یک چالش مهم است. در همین راستا به منظور مدیریت مسئله و بهینه سازی فرایند زمانبندی، از راهکارهای یادگیری عمیق استفاده می شود. با توجه به این که، در روش پیشنهادی یک رویکرد زمانبندی بهینه ارائه می شود که وابسته به پارامترهای مختلف کیفیت سرویس است، لذا می توان از این پارامترها به عنوان مولفه­های ارزیابی در راهکار پیشنهادی استفاده نمود.

در روش پیشنهادی می توان از پارامترهای کیفیت سرویس مختلفی از قبیل پهنای باند، نرخ گذردهی، نرخ خطای بیتی، قابلیت اطمینان، قابلیت اعتماد، هزینه و موارد مشابه دیگر برای بهینه سازی انتخاب منابع جهت زمانبندی کارا در محیط لبه و مه استفاده کرد، ولی باید دقت داشت که استفاده از پارامترهای بیشتر، ­می­تواند منجر به افزایش پیچیدگی محاسباتی و در نتیجه افت کارایی کلی شبکه شود. با توجه به این که در این روش از پارامترهای کیفیت سرویس چندگانه استفاده می شود. نیاز به نرمال سازی این پارامترها و همچنین حذف افزونگی اطلاعات وجود دارد. در راهکار پیشنهادی برای نرمال سازی پارامترهای کیفیت سرویس، از فرایند نرمال­سازی استاندارد استفاده شده و به منظور حذف افزونگی از مدل پیرسون استفاده می شود. در شکل زیر ساختار راهکار پیشنهادی شرح داده شده است.

شکل 3- 1) مدل مفهومی رویکرد پیشنهادی

# 4-3) الگوریتم‌ها و راهکارها

در این بخش از پژوهش، الگوریتم ها و راهکارای ارائه شده در پژوهش شامل پیش پردازش و نرمال سازی داده های کیفیت سرویس، توصیف خواهد شد. در ادامه نیز به بررسی و ارزیابی فرایند اجرایی مدل پیشنهادی پرداخته می شود.

# 1-4-3) پارامترهای کیفیت سرویس

در این بخش از پژوهش، پارامترهای کیفیت منابع پردازشی که به منظور اجرای فرایند تخلیه منابع مورد استفاده قرار می­گیرند، توصیف خواهند شد در این پژوهش، از پارامترهای کیفیت سرویس، نرخ خطای بیتی منابع پردازشی، قابلیت اطمینان، پهنای باند مصرفی و سرعت پردازشی منابع استفاده می شود که در ادامه هریک از این پارامترها توصیف می شوند.

* **پهنای­باند**: پهنای باند یکی از پارامترهای اولیه برای زمانبندی کارها، در تعیین اولویت واگذاری منابع در سیستم های رایانش ابری است که علاوه بر قابلیت های محاسباتی منابع، قابلیت های ارتباطی را نیز در نظر می­گیرد. در الگوریتم های زمانبندی کار، منابع بر اساس نوع الگوریتم باتوجه به مقدار پهنای باند می­توانند به کارهای مختلف واگذار شوند. پهنای باند برابر نرخ انتقال داده در واحد زمان توسط یک اتصال شبکه است. واحد اندازه­گیری این پارامتر مگابیت بر ثانیه در نظر گرفته می شود. اصطلاح پهنای باند معانی فنی متعددی دارد، اما از زمان رواج اینترنت، عموماً به حجم اطلاعات در واحد زمان اشاره دارد که یک رسانه انتقال (مانند اتصال اینترنت) می تواند از عهده آن برآید. یک اتصال اینترنتی با پهنای باند بیشتر می تواند مقدار مشخصی از داده (مثلاً یک فایل ویدیویی) را بسیار سریعتر از اتصال اینترنتی با پهنای باند کمتر منتقل کند. تعریف دیگر از پهنای باند در حوزه ارتباطات بی سیم، اشاره به پهنای باند سیگنال آنالوگ اندازه گیری شده بر حسب هرتز، بین کمترین و بالاترین فرکانس قابل دستیابی استفاده می شود. در یک شبکه کامپیوتری، حداکثر پهنای باند از رابطه 3-1 به دست می آید[34]:

|  |  |
| --- | --- |
| (3-1) |  |

که در این رابطه، RWINپنجره دریافت TCPو RTT زمان رفت و برگشت[[4]](#footnote-4) در منبع پردازشی است.

* **قابلیت اطمینان**: هنگام طراحی و توسعه یک محیط رایانش ابری، ارائه روشی با قابلیت اطمینان بالا در این شبکه ها برای ارسال دقیق و سریع پیام ها بسیار مهم است. به منظور بهبود قابلیت اطمینان در شبکه، تکنیک های مختلفی به عنوان راه حلی برای بهبود تحمل خطا در این سیستم­ها معرفی شده است. تحمل خطا شامل شناسایی خطاها و خرابی ها در منبع پردازشی و بازیابی خرابی ها برای ادامه ارسال داده ها است. یکی از مهمترین معیارها در تصمیم­گیری، برنامه ریزی مهندسی، نگهداری و تعمیر، موضوع قابلیت اطمینان است. تعریف کلی قابلیت اطمینان را می توان به صورت زیر در نظر گرفت: قابلیت اطمینان یک منبع پردازشی در ساختار ابر و مه این امکان را دارد که در آن منبع پردازشی برای مدت معینی در شرایط موجود و از پیش تعیین شده به صورت سالم و بدون نقص فعالیت کند. به طور کلی، بسته به موقعیت استفاده و کاربرد، تعاریف مختلفی از قابلیت اطمینان وجود دارد. مشترک همه این تعاریف این است که تابع قابلیت اطمینان R(t) نشان دهنده احتمال عملکرد بهینه منبع پردازشی در بازه (0, T) است. واضح است که در حالت کلی بهترین عملکرد در زمان t به این معنی است که واحد مورد نظر در دوره (0, T) بهترین کارایی را داشته است، مشروط بر اینکه هیچ قطعه­ای در این مدت تعویض یا تعمیر نشده باشد. قابلیت اطمینان در طول زمان تغییر می کند، بنابراین قابلیت اطمینان را می توان به عنوان تابعی از زمان به صورت R(t) بیان کرد. تابع قابلیت اطمینان مکمل تابع توزیع احتمال شکست است، بنابراین می توان آن را با تابع چگالی احتمال تعریف کرد[34].

از آنجایی که تابع قابلیت اطمینان نشان دهنده احتمال شکست است، این توابع به عنوان توابع پایداری نیز شناخته می شوند. زیرا این توابع احتمال عمر باقیمانده تجهیزات را تعیین می­کنند. به طور کلی قابلیت اطمینان یک منبع پردازشی R برابر با احتمال این است که این منبع در زمان معین و در شرایط خاص ویژگی های مورد نیاز را داشته باشد و در غیر این صورت، عدم اطمینان F برابر است با احتمال اینکه موجودیت مورد نظر نتواند ویژگی های خواسته شده را در بازه زمانی معین تحت شرایط کاری به دست آورد. عدم اطمینان و قابلیت اطمینان هردو وابسته به زمان هستند. در زمان صفر، قابلیت اطمینان موجودیتی که آغاز به فعالیت می کند برابر یک است، پس از مدت زمانی این مقدار به 0.5 و سپس زمانی که از کار بیفتد به صفر می رسد. به بیان دیگر، عدم اطمینان از مقدار صفر شروع شده و به طور صعودی عمل کرده و زمانی که سیستم خراب می شود به مقدار یک می رسد. به طورکلی در هر لحظه مجموع قابلیت اطمینان و عدم اطمینان برابر با یک خواهد بود. این مسئله را می­توان با استفاده از رابطه 3-2 بیان نمود [38].

|  |  |
| --- | --- |
| (3-2) |  |

قابل قبول ترین تعریف برای قابلیت اطمینان برابر است با توانایی یک موجودیت (محصول، سیستم، و غیره) در انجام وظایف تحت شرایط عملیاتی طراحی شده برای مدت زمان معین. بنابراین قابلیت اطمینان را می توان با استفاده از رابطه 3-3 تعریف نمود [38]:

|  |  |
| --- | --- |
| (3-3) |  |

که در این رابطه نشان دهنده مدت زمان معین یا سیکل کاری موجودیت است و نشان دهنده زمان شکست است و نشان دهنده قابلیت اطمینان موجودیت است. در این رابطه نشان دهنده شرایط تعیین شده همچون شرایط محیطی هستند، معمولاً این شرایط در آنالیز قابلیت اطمینان نادیده گرفته می شوند و معادله 3-3 را می توان به صورت معادله 3-4 بازنویسی نمود[38].

|  |  |
| --- | --- |
| (3-4) |  |

* **نرخ پردازشی منابع:** با توسعه فناوری، افزایش اهداف بهره وری، اینترنت سریع تر و دستگاه های بیشتر، نیاز به سرعت پردازشی افزایش یافته است. پردازنده های رایانه و سرعت ساعت آن ها ویژگی است که معمولاً با فناوری با عملکرد بالا و سریع مرتبط هستند. سرعت پردازنده یک منبع یکی از مهمترین عناصری است که باید در هنگام مقایسه منابع پردازشی در نظر گرفت. منابع پردازشی اغلب به عنوان "مغز" یک سیستم شناخته می شود، بنابراین اطمینان از عملکرد صحیح آن برای افزایش طول عمر شبکه و بهبود فرایندهای عملیاتی بسیار مهم است.
* **نرخ گذردهی:** نرخ گذردهی در شبکه های نرم افزار محور، برابر متوسط نرخ تحویل موفق پیام در یک کانال ارتباطی است. این داده ها ممکن است از یک پیوند فیزیکی، منطقی یا با عبور از طریق [گره های](https://fa.wikipedia.org/wiki/%DA%AF%D8%B1%D9%87_%D8%B4%D8%A8%DA%A9%D9%87) خاص شبکه، تحویل داده شوند. نرخ گذردهی معمولاً به وسیله [بیت](https://fa.wikipedia.org/wiki/%D8%A8%DB%8C%D8%AA_(%D8%B1%D8%A7%DB%8C%D8%A7%D9%86%D9%87)) بر ثانیه و در برخی موارد [بسته های داده](https://fa.wikipedia.org/w/index.php?title=%D8%A8%D8%B3%D8%AA%D9%87%E2%80%8C%D9%87%D8%A7%DB%8C_%D8%AF%D8%A7%D8%AF%D9%87&action=edit&redlink=1) بر ثانیه در [بازه های مختلف زمانی](https://fa.wikipedia.org/w/index.php?title=%D8%A8%D8%A7%D8%B2%D9%87%E2%80%8C%D9%87%D8%A7%DB%8C_%D8%B2%D9%85%D8%A7%D9%86%DB%8C&action=edit&redlink=1)، [اندازه گیری](https://fa.wikipedia.org/wiki/%D8%A7%D9%86%D8%AF%D8%A7%D8%B2%D9%87%E2%80%8C%DA%AF%DB%8C%D8%B1%DB%8C) می شود. در کل نرخ گذردهی عبارت است از مجموع نرخ داده هایی که به همه پایانه ها در یک شبکه تحویل داده شده است. نرخ گذردهی را می توان به صورت ریاضی با استفاده از [تئوری صف](https://fa.wikipedia.org/wiki/%D8%AA%D8%A6%D9%88%D8%B1%DB%8C_%D8%B5%D9%81) توصیف نمود که در آن، بار در بسته در واحد زمان با نرخ ورود مشخص می شود و توان در بسته در هر واحد زمان با نرخ خروج مشخص می شود. در کل نرخ گذردهی یک شبکه را می توان با استفاده از ابزارهای مختلف موجود بر روی سیستم عامل های مختلف اندازه گیری نمود.
* **فاصله ارتباطی اشیا:** در ساختار شبکه های ارتباطی، توابع زیادی برای اندازه گیری فاصله بین گره ها در شبکه وجود دارد. توابع فاصله ای کاربردهای زیادی در تکنیک ارتباطی شبکه های مختلف به ویژه در گره های عملیاتی دارند. در این کار، تابع فاصله اقلیدسی برای محاسبه فاصله بین دو گره در یک شبکه استفاده شد. با استفاده از فاصله اقلیدسی، کوتاه ترین فاصله بین دو نقطه بر اساس رابطه فیثاغورث محاسبه می شود. اگر برابر نقطه قرار گیری سرخوشه و برابر نقطه قرارگیری چاهک باشد، فاصله اقلیدسی بین این دو نقطه را می توان با استفاده از رابطه 3-5 محاسبه کرد[28].

|  |  |
| --- | --- |
| *(3-5)* |  |

*که در آن شروط زیر برقرار هستند:*

* *فاصله اقلیدسی باید نامنفی باشد: مشخص است که این رابطه مقداری نامنفی دارد زیرا از مجموع مربعات تفاضل ها ساخته شده است.*
* *وجود رابطه انعکاسی برای فاصله اقلیدسی: برای نقطه های و با مقدار مولفه های یکسان، فاصله اقلیدسی برابر با صفر خواهد بود. همچنین زمانی که فاصله اقلیدسی صفر باشد باید همه مربعات تفاضل ها صفر باشند، در نتیجه دو نقطه بر هم منطبق هستند.*

# 2-4-3) پیش پردازش اطلاعات

برای دستیابی به داده های قابل استفاده جهت تصمیم گیری بر اساس پارامترهای کیفیت سرویس بیان شده در بخش قبل، ابتدا باید داده های مختلف ارائه شده همبسته سازی شوند تا بتوان داده های مفید را در قالب یک مدل کلی مشخص نموده و داده های زائد که نقشی در تصمیم گیری ندارند را حذف نمود. به منظور همبسته سازی داده ها، روش های بسیار متنوعی وجود دارد؛ اساس تمامی این روش ها تعیین ضریب همبستگی بین پارامترهای مختلف است. در این پژوهش برای تعیین ضریب همبستگی داده ها از آزمون همبستگی پیرسون[[5]](#footnote-5) استفاده خواهد شد و در ادامه نتایج به دست آمده از این آزمون به عنوان پارامترهای ورودی برای تعیین منابع بهینه استفاده می شوند. ضریب همبستگی[[6]](#footnote-6) ابزاری آماری برای تعیین نوع و درجه رابطه یک متغیر کمی با متغیر کمی دیگر است. ضریب همبستگی، یکی از معیارهای مورد استفاده در تعیین همبستگی دو متغیر است. ضریب همبستگی شدت رابطه و همچنین نوع رابطه (مستقیم یا معکوس) را نشان می دهد. این ضریب بین ۱ تا ۱- است و در عدم وجود رابطه بین دو متغیر، برابر صفر است. احتمالاً، گسترده ترین کاربرد شاخص آماری همبستگی دو متغیری، ضریب همبستگی گشتاوری پیرسون است که به طورمعمول همبستگی پیرسون نامیده می شود. علامت اختصاری آن است. ضریب پیرسون نشان می دهد که تا چه اندازه بین متغیرهای کمّی رابطه خطی وجود دارد. کاربرد اصلی ضریب همبستگی پیرسون زمانی است که متغیرها از نوع پارامتری باشند؛ بدین معنا که توزیع نرمال داشته باشند و در سطح فاصله ای/ نسبی باشند. البته زمانی که متغیرها از نوع شبه فاصله ای باشند (یعنی هر متغیر ترکیبی از چند متغیر ترتیبی باشد که اصطلاحاً به آن مقیاس های تراکمی می­گویند). درحالت کلی ضریب همبستگی پیرسون بین دو متغیر تصادفی برابر با [کوواریانس](https://fa.wikipedia.org/wiki/%DA%A9%D9%88%D9%88%D8%A7%D8%B1%DB%8C%D8%A7%D9%86%D8%B3) آن ها تقسیم بر [انحراف معیار](https://fa.wikipedia.org/wiki/%D8%A7%D9%86%D8%AD%D8%B1%D8%A7%D9%81_%D9%85%D8%B9%DB%8C%D8%A7%D8%B1) آن ها است. برای یک جامعه آماری، ضریب همبستگی جامعه به صورت رابطه (3-6) قابل تعریف خواهد بود[35].

|  |  |
| --- | --- |
| (6-3) |  |

که در این رابطه cov، نشان دهنده کوواریانس است و نشان دهنده انحراف معیار متغیر X است و نشان دهنده انحراف معیار متغیر Y است. نشان دهنده میانگین متغیر X است و نشان دهند میانگین متغیر Y است. نهایت E نشان دهنده امید به ریاضی است. در حالت کلی ضریب همبستگی پیرسون برای یک نمونه آماری با n زوج داده به صورت با استفاده از رابطه (3-7) تعریف می شود[35].

|  |  |
| --- | --- |
| (7-3) |  |

رابطه بالا را می توان به صورت خلاصه تر با استفاده از رابطه (3-8) بیان نمود[35].

|  |  |
| --- | --- |
| (8-3) |  |

که در آن هر یک از مقادیر،،و به ترتیب با استفاده از روابط (3-9)، (3-10)، (3-11) و (3-12) تعریف می شوند[35].

|  |  |
| --- | --- |
| (9-3) |  |
| (10-3) |  |
| (11-3) |  |
| (12-3) |  |

بعد از تعیین معنی داری و جهت رابطه، باید شدت رابطه ارزیابی شود. برای تفسیر شدت رابطه دو متغیر، بسته به کاربردهای موردنظر، تقسیم بندی های گوناگونی ارائه شده­ است. از این تقسیم بندی ها به منظور همبسته سازی داده ها و حذف داده های غیرمفید از بین انبوهی از داده ها استفاده می شود. با استفاده از همبسته نمودن داده ها می توان بخش بسیار بزرگی از ارزیابی های غیرمفید برای داده های غیرضروری را حذف نمود. تقسیم بندی ارائه شده در جدول 3-1 نمونه­ای از این تقسیم­بندی است[35].

**جدول 3- 1) شیوه تفسیر شدت رابطه در همبستگی پیرسون**

|  |  |
| --- | --- |
| شدّت رابطه | تفسیر |
| 8/. تا 1 | **رابطه بسیار قوی** |
| 6/. تا 8/. | **رابطه قوی** |
| 4/. تا 6/. | **رابطه متوسط** |
| 2/. تا 4/. | **رابطه کم (یا ضعیف)** |
| صفر تا 2/. | **فقدان رابطه یا رابطه ناچیز** |

# 3-4-3) نرمال‌سازی اطلاعات

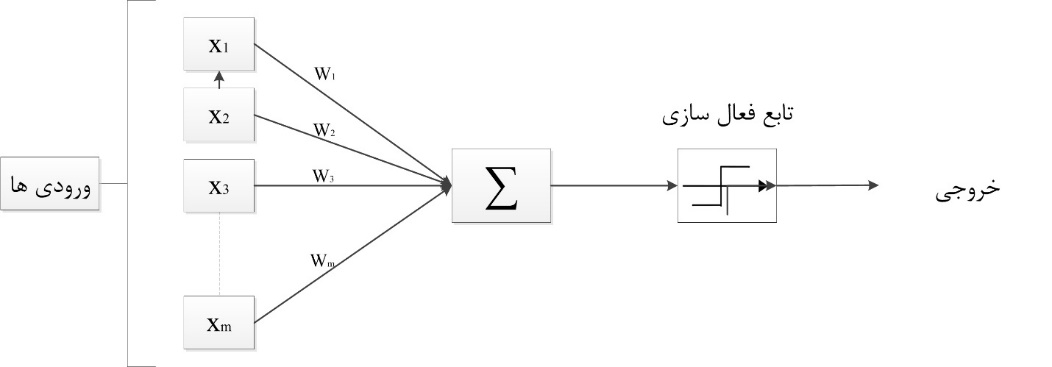
نرمال سازی داده ها روشی برای یکنواخت کردن بازه مقادیر مربوط به متغیرهای مختلف پژوهش است و به بی مقیاس سازی داده ها نیز معروف است. اگر واحد سنجش متغیرهای مورد مطالعه متنوع باشد با استفاده از روش های نرمال سازی می توان داده ها را بی مقیاس کرد. در این پژوهش به منظور بی مقیاس کردن پارامترهای کیفیت سرویس از روش نرمال سازی استفاده می شود. یک روش ساده برای نرمال کردن اعداد مطرح شده که به محاسبه بردار ویژه نیز معروف شده است. در این روش کافی است هر عدد در یک مجموعه بر مجموع عناصر آن مجموعه تقسیم شود. در این صورت جمع کل عناصر پس از نرمال سازی یک خواهد بود. در رابطه 3-13 فرایند نرمال سازی اطلاعات به منظور بی مقیاس کردن داده ها ارائه شده است.

|  |  |
| --- | --- |
| (13-3) |  |

*که در این رابطه نشان دهنده پارامتر کیفیت سرویس i ام برای گره jام و نشان دهنده مجموع مقادیر پارامترهای کیفیت سرویس برای گره j ام است هست.*

# 4-4-3) شبکه عصبی LSTM

به طورکلی، شبکه عصبی یک سامانه پردازشی داده است که از مغز انسان الهام گرفته و پردازش داده ها را بر عهده پردازنده های کوچک و بسیار زیادی سپرده که به صورت شبکه ای به هم پیوسته و موازی با یکدیگر رفتار می کنند تا یک مسئله را حل نمایند در يك نگاه ساده، مدل يك عصب بايد شامل ورودی هایی باشد كه در نقش سيناپس انجام وظیفه كنند. اين ورودی ها در وزن نهايي ضرب می شوند تا قدرت سيگنال را تعيين كنند. نهايتاً يك عملگر رياضي، تصميم­گيري می کند كه آيا نورون فعال شود يا خير و اگر جواب مثبت باشد، ميزان خروجي را مشخص می سازد. بنابراين شبكه عصبي مصنوعي با استفاده از مدل ساده شده عصب واقعي به پردازش اطلاعات می پردازد. باتوجه به اين توضيحات، می توان مدل ساده­اي براي توصيف يك نورون (يك گره در شبكه عصبي مصنوعي) پيشنهاد كرد. اين مدل در شكل 3-3 نشان داده شده است.



**شکل 3- 2) مدل آموزش در شبکه عصبی**

همان طور که مشاهده می شود ورودی ها که شامل مولفه های مربوط به ساختار شبکه عصبی هستند، در تصویر بالا با مقادیر نمایش داده می شوند. در مدل شبکه عصبی مصنوعی به هر ورودی یک وزن اختصاص داده می شود. این وزن ها در واقع اهمیت ورودی ها را تعیین می کنند، یعنی هر چه وزن بیشتر باشد، ورودی برای آموزش شبکه مهم تر است. سپس تمامی ورودی ها با هم جمع شده و به صورت یک لایه به آکسون وارد می شوند. در مرحله بعد تابع فعال سازی[[7]](#footnote-7) بر روی داده ها اعمال می شود. تابع فعال سازی در واقع نسبت به نیاز مسئله و نوع شبکه عصبی تعریف می شود. این تابع شامل یک فرمول ریاضی برای به روزرسانی وزن ها در شبکه است که می توان با استفاده از رابطه 3-14 بیان نمود.

|  |  |
| --- | --- |
| (3-14) |  |

که در آن پارامتر تنظیم کننده در ساختار شبکه اینترنت اشیا مبتنی بر مدیریت ساختار مصنوعی است. پس از انجام محاسبات در این مرحله اطلاعات از طریق سیناپس های خروجی وارد نورون دیگر می شوند و این مرحله تا جایی ادامه پیدا می کند که شبکه اصطلاحاً آموزش کامل ببیند. در مدل ارائه شده، تنوع های بسياري وجود دارد.

وزن های گره های موجود در اینترنت اشیا که به صورت نودهای پردازشی شبکه در نظر گرفته می شوند، می توانند مثبت يا منفي باشند. از طرفي، توابع مورد استفاده براي آستانه گذاری می توانند بسيار متنوع باشند. از جمله مشهورترين اين توابع مي­توان به تابع هایی نظير arcsin. arctan. sigmoid اشاره كرد. اين توابع بايد پيوسته و هموار بوده و مشتق پذیر باشند. همچنين تعداد گره های ورودی می تواند متغير باشند. البته زیاد شدن تعداد اين گره ها، به وضوح تعيين وزن ها را با مشكل روبرو می کند؛ لذا بايد به دنبال روش هایی براي حل اين موضوع بود. روند تعين وزن های بهينه و تنظيم مقادير آن ها عمدتاً به صورت بازگشتي انجام می شود. بدين منظور شبكه را با استفاده از قواعد و داده ها آموزش داده و با استفاده از قابليت يادگيري شبكه، الگوریتم های متنوعي پیشنهاد می شود كه همگي سعي در نزدیک کردن خروجي تولید شده توسط شبكه به خروجي ایده آل و مورد انتظار را دارند. هرچند نحوه مدل کردن نورون جزء اساسی ترین نكات در کارایی شبكه عصبی است، اما نحوه برقراري اتصالات و چيدمان (توپولوژي) شبكه نيز فاكتور بسيار مهم و اثرگذاري است. بايد توجه داشت كه توپولوژي مغز انسان آن قدر پيچيده است كه نمی توان از آن به عنوان مدلي براي اعمال به شبكه عصبي استفاده نمود، چرا كه مدلي كه استفاده می شود، يك مدل ساده شده است درحالی که چيدمان مغز از المان های بسيار زيادي استفاده می کند.

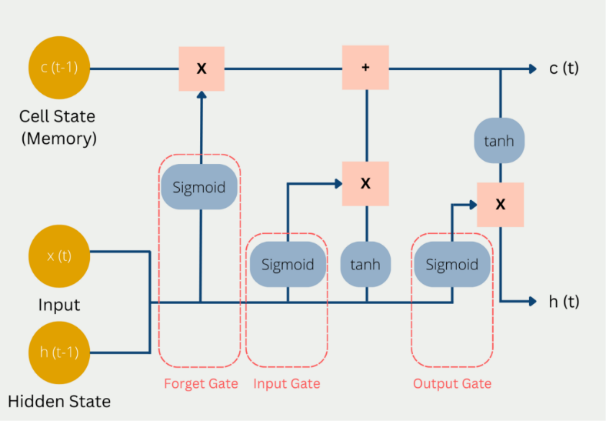
شبکه های عصبی LSTM به طور خاص برای غلبه بر مشکل وابستگی طولانی مدت با شبکه های عصبی مکرر (به دلیل مشکل گرادیان ناپدیدشدن) طراحی شده اند. LSTM ها دارای اتصالات بازخوردی هستند که آنها را با شبکه های عصبی پیش خور سنتی متفاوت می کند. این ویژگی LSTM ها را قادر می­سازد تا کل توالی داده ها (به عنوان مثال سری های زمانی) را بدون رسیدگی به هر نقطه در دنباله به طور مستقل پردازش کنند، بلکه اطلاعات مفیدی در مورد داده های قبلی در توالی برای کمک به پردازش نقاط داده جدید حفظ کنند. در نتیجه، LSTM ها به ویژه در پردازش توالی داده ها مانند متن، گفتار و سری های زمانی عمومی خوب هستند.

بر اساس این الگوریتم، در سطح پایه، خروجی یک LSTM در یک نقطه زمانی خاص به سه چیز وابسته است:

* حافظه بلند مدت فعلی شبکه - که به صورت وضعیت سلولی شناخته می شود
* خروجی در نقطه قبلی در زمان - به صورت حالت پنهان قبلی شناخته می شود
* داده های ورودی در مرحله زمانی فعلی

LSTM ها از یک سری "دروازه" استفاده می کنند که نحوه ورود، ذخیره و خروج اطلاعات در یک دنباله داده از شبکه را کنترل می کند. سه دروازه در یک LSTM معمولی وجود دارد. گیت فراموشی، گیت ورودی و گیت خروجی. این گیت ها را می توان فیلتر در نظر گرفت و هر کدام شبکه عصبی خودشان هستند.

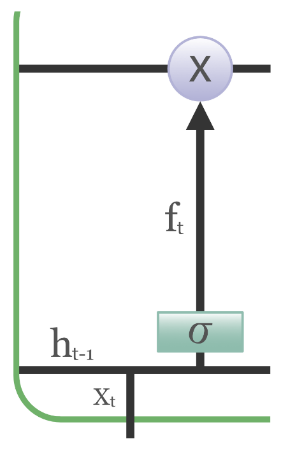
مشکل شبکه های عصبی بازگشتی این است که آنها به سادگی داده های قبلی را در "حافظه کوتاه مدت" خود ذخیره می کنند. هنگامی که حافظه موجود در آن تمام شد، به سادگی طولانی ترین اطلاعات حفظ شده را حذف می کند و آن را با داده های جدید جایگزین می کند. مدل LSTM سعی می کند با حفظ اطلاعات انتخاب شده در حافظه بلند مدت این مشکل را رفع کند. این حافظه طولانی مدت در حالت سلولی ذخیره می شود. باتوجه به این مسئله، در روش پیشنهادی برای ارائه یک راهکار کیفیت سرویس، در هر مرحله محاسباتی، ورودی فعلی ، که شامل پارامترهای موثر در کیفیت سرویس است، حالت قبلی حافظه کوتاه مدت برابر و حالت قبلی حالت پنهان استفاده می شود. ساختار اولویت بندی مبتنی بر رویکرد پیشنهادی در شکل 3-3 نشان داده شده است:



**شکل 3- 3) معماری اولویت‌بندی پارامترهای کیفیت سرویس مبتنی بر شبکه عصبی LSTM**

در راهکار پیشنهادی، به منظور طبقه­بندی توصیه ها به کاربران سه پارامتر ، و در مسیر خود از گیت­های زیر عبور می کنند:

* در گیت فراموشی[[8]](#footnote-8)، تصمیم گرفته می شود که کدام اطلاعات فعلی و قبلی در مورد پارامترهای کیفیت سرویس برای هر توصیه نگهداری شود و کدام یک به بیرون استخراج شود. این شامل وضعیت پنهان از پاس قبلی و ورودی فعلی است. این مقادیر به یک تابع سیگموئید[[9]](#footnote-9) منتقل می شوند که فقط می تواند مقادیر بین 0 و 1 را خروجی دهد. مقدار 0 به این معنی است که اطلاعات قبلی را می توان فراموش کرد؛ زیرا احتمالاً اطلاعات جدید و مهم تری وجود دارد. عدد یک به این معنی است که اطلاعات قبلی حفظ می شود. ساختار گیت فراموشی باتوجه به شکل 4-3 قابل توصیف است:



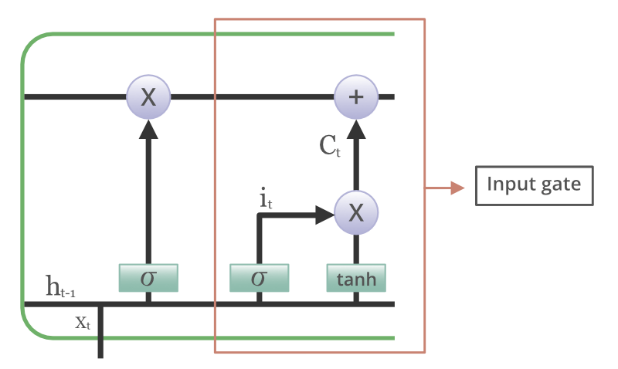
**شکل 3- 4) ساختار گیت فراموشی مبتنی بر شبکه عصبی LSTM**

نتایج حاصل از این در حالت سلولی فعلی ضرب می شود، بنابراین دانشی که دیگر موردنیاز نیست فراموش می شود، زیرا در 0 ضرب می شود و در نتیجه حذف می شود. رابطه مربوط به گیت فراموشی بر اساس رابطه 3-15 قابل توصیف است:

|  |  |
| --- | --- |
| (3-15) |  |

جایی که در آن:

* نشان دهنده ماتریس وزن مرتبط با دروازه فراموشی است.
* نشان دهنده پیوستگی ورودی فعلی و حالت پنهان قبلی است.
* سوگیری با گیت فراموشی است.
* σ تابع فعال سازی سیگموئید است.
* در گیت ورودی[[10]](#footnote-10)، تصمیم گرفته می شود که ورودی فعلی چقدر برای حل کار ارزشمند است. لذا، ورودی فعلی در حالت پنهان و ماتریس وزن آخرین اجرا ضرب می شود. سپس تمام اطلاعاتی که در ورودی مهم به نظر می رسند به حالت سلول اضافه می شوند و حالت سلولی جدید را تشکیل می دهند. این وضعیت سلولی جدید اکنون وضعیت فعلی حافظه بلند مدت است و در اجرای بعدی مورد استفاده قرار خواهد گرفت. همچنین، افزودن اطلاعات مفید به حالت سلول توسط گیت ورودی انجام می شود. ساختار گیت ورودی در شکل 3-5 نشان داده شده است:



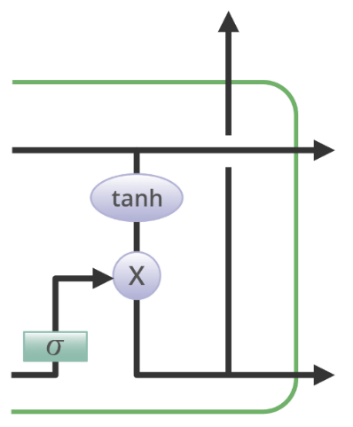
**شکل 3- 5) ساختار گیت ورودی مبتنی بر شبکه عصبی ­LSTM**

در این گیت، ابتدا، اطلاعات با استفاده از تابع sigmoid تنظیم می شود و مقادیری که باید به خاطر سپرده شوند، مشابه دروازه فراموشی با استفاده از ورودی های و فیلتر می شوند. سپس، یک بردار با استفاده از تابع tanh ایجاد می شود که خروجی از 1- تا 1+ می دهد، که شامل تمام مقادیر ممکن از و است. در نهایت، مقادیر بردار و مقادیر تنظیم شده برای به دست آوردن اطلاعات مفید ضرب می شوند. معادله گیت ورودی بر اساس رابطه3-16 قابل توصیف است:

|  |  |
| --- | --- |
| (3-16) | = tanh ()  = ⊙ C)t-1) + ⊙ |

جایی که در آن:

* ⊙ نشان دهنده ضرب عنصر است
* tanh تابع فعال سازی tanh است
* گیت خروجی[[11]](#footnote-11)، خروجی مدل LSTM در حالت پنهان محاسبه می شود. بسته به کاربرد، می تواند؛ مثلاً کلمه ای باشد که معنای جمله را تکمیل کند. برای انجام این کار، تابع سیگموئید تصمیم می گیرد که چه اطلاعاتی می تواند وارد گیت خروجی شود و سپس حالت سلول پس از فعال شدن با تابع tanh ضرب می شود. همچنین وظیفه استخراج اطلاعات مفید از وضعیت سلول فعلی برای ارائه به عنوان خروجی توسط گیت خروجی انجام می شود. ابتدا یک بردار با اعمال تابع tanh بر روی سلول تولید می شود. سپس، اطلاعات با استفاده از تابع sigmoid و فیلتر توسط مقادیری که باید با استفاده از ورودی های و به خاطر سپرده شوند، تنظیم می شوند. ساختار گیت خروجی در شکل 6-3 نشان داده شده است.



**شکل 3- 6) ساختار گیت خروجی مبتنی بر شبکه عصبی ­LSTM**

در نهایت، مقادیر بردار و مقادیر تنظیم شده ضرب می شوند تا به عنوان خروجی و ورودی به سلول بعدی ارسال شوند. معادله گیت خروجی به صورت رابطه 3-17است:

|  |  |
| --- | --- |
| (3-17) | = σ() |

در نهایت وقتی به روزرسانی ها در حافظه بلند مدت شبکه کامل شده است، می توان به مرحله نهایی که نشان دهنده تصمیم گیری در مورد وضعیت پنهان جدید است حرکت کرد. برای تصمیم گیری در این مورد، از سه مؤلفه استفاده می شود. وضعیت سلولی که به تازگی به روز شده است، وضعیت پنهان قبلی و داده های ورودی جدید.

# 5-3) زمانبندی وظایف با استفاده از روش پیشنهادی

حال با نرمال سازی و تعیین همبستگی مقادیر پارامترهای کیفیت سرویس هر یک از منابع، از مدل شبکه عصبی LSTM به منظور طبقه بندی منابع برای حل مسئله استفاده می شود. با توجه به اینکه هر یک از پارامترهای کیفیت بیان شده در بالا، دارای واحدهای مختلفی بوده و نمی توان آن ها را با یکدیگر مقایسه کرد، در نتیجه باید از یک روش نرمال سازی برای نرمال نمودن مقادیر و در نتیجه قابل مقایسه کردن این پارامترها استفاده شود. در نهایت یک ماتریس m در n به منظور ارزیابی هریک از منابع به دست می آید. این ماتریس شامل یک ترتیب نزولی از منابع مناسب (بر اساس پارامترهای کیفیت سرویس) برای تخلیه بار در شبکه است. گزینش، منبعی با پارامترهای کیفیت سرویس مناسب، که دارای قابلیت اطمینان، نرخ خطای بیتی و نرخ گذردهی بالا و فاصله بین گره های کمتری هستند، می تواند منجر به ایجاد تعادل بار در شبکه شده و کارایی را افزایش دهد. علاوه بر این روش پیشنهادی با کاهش ترافیک در شبکه نرخ تحویل موفق بسته را افزایش داده و تأخیر را به صورت چشمگیری کاهش می­دهد. بعد از اتمام هر دور از تخصیص وظایف در شبکه، مقادیر هر یک از پارامترهای بیان شده برای منابع به­روزرسانی شده و مراحل فوق برای دور بعدی زمانبندی وظایف تکرار می شوند. در ادامه الگوریتم روش پیشنهادی به صورت مرحله به مرحله توصیف می شود.

# 6-3) فلوچارت روش پیشنهادی

به‌طور کلی فلوچارت روش پیشنهادی در شکل 3-7 نشان داده‌شده‌است.



شکل 3- 7) فلوچارت رویکرد پیشنهادی

# 7-3) الگوریتم نهایی

در این بخش از پژوهش، الگوریتم نهایی رویکرد پیشنهادی به صورت گام به گام شرح داده شده و در ادامه فلوچارت روش پیشنهادی بیان می شود.

گام اول) شروع

گام دوم) تعیین جمعیت اولیه منابع پردازشی

گام سوم) مقدار دهی اولیه پارامترهای کیفیت سرویس برای منابع موجود

گام چهارم) ارزیابی پارامترهای کیفیت سرویس برای هریک از درخواست­ها و منابع به منظور بارگذاری وظایف

گام پنجم) نرمال سازی پارامترهای کیفیت سرویس به منظور یکسان کردن مقیاس داده ها

گام ششم) ورود داده ها در گیت فراموشی شبکه عصبی LSTM

گام هفتم) انتخاب داده های مورد نظر با استفاده از گیت فراموشی و انتقال آنها به گیت شبکه عصبی LSTM

گام هشتم) اجرای تابع سیگموئید به منظور طبقه بندی منابع پردازشی بر اساس ورودی های کیفیت سرویس

گام نهم) ارسال داده ها به گیت خروجی شبکه عصبی LSTM به منظور تعیین طبقه منابع پردازشی

گام دهم) اجرای فرایند زمانبندی و تخصیص منابع

گام یازدهم) به روز رسانی پارامترهای کیفیت سرویس پس از بارگذرای منابع پردازشی

گام دوازدهم) تکرار مراحل چهارم تا دهم تا پایان بارگذاری وظایف

گام سیزدهم) پایان

# 8-3) جمع بندی فصل

در این فصل از پژوهش، رویکرد پیشنهادی زمانبندی در محیط های محاسباتی لبه و مه مبتنی بر یادگیری عمیق با استفاده از شبکه عصبی حافظه طولانی کوتاه مدت برای بهینه سازی بار سیستم و زمان پاسخ ارائه شده و الگوریتم های بکارگرفته شده در آن به همراه ساختار کلی روش پیشنهادی توصیف گردید. در ادامه، فلوچارت و الگوریتم روش پیشنهادی نیز توصیف گردید. در فصل آتی به بررسی و ارزیابی کارایی روش پیشنهادی در مقایسه با رویکردهای مشابه بررسی شده و نتایج ارزیابی به صورت جدول و نمودار توصیف می شوند.

4) فصل چهارم

**ارزیابی نتایج**

**اهداف فصل:**

* توصیف نتایج شبیه سازی
* ارزیابی نتایج شبیه سازی
* مقایسه روش پیشنهادی با رویکردهای موجود

# 1-4) مقدمه

در این فصل از پژوهش، شبیه سازی ها برای ارزیابی عملکرد روش پیشنهادی انجام شده است. در ابتدا، محیط شبیه سازی، روش مدیریت اطلاعات، برنامه های مرجع و سناریوهای شبیه سازی شرح داده شده و در ادامه روش پیشنهادی با رویکردهای مشابه در این زمینه مقایسه خواهد شد. سپس نتایج ارزیابی از دیدگاه‌های مختلف ارائه و مورد بحث قرار می گیرد. در ادامه توضیح مختصری در مورد محیط شبیه سازی با عنوان ifogsim ارائه شده و در ادامه فرایند شبیه سازی و نتایج به‌دست‌آمده بیان می شوند.

# 2-4) معرفی محیط شبیه سازی

محاسبات ابری قبلاً ثابت کرده است که یک ساختار بهینه برای نسل بعدی سیستم‌های فیزیکی-سایبری است. حجم قابل توجهی از داده های تولید شده توسط دستگاه های مجازی در ساختار محاسبات ابری نیاز به پردازش حساس به تأخیر دارد که با استقرار برنامه های مربوطه در مراکز داده ابری راه دور امکان پذیر نیست. زیرساخت مقیاس پذیر محاسبات ابری یا محیط مه یا لبه برای اعتبار بخشیدن به این سیاست ها ضروری است، که به دلیل هزینه زیاد و زمان اجرای آن در دنیای واقعی نیز چالش برانگیز است. با در نظر گرفتن شبیه سازی به عنوان یک کلید برای این محدودیت، نرم افزارهای مختلفی توسعه یافته اند که می‌توانند رفتار فیزیکی محیط های محاسباتی مه یا لبه را تقلید کنند. با این وجود، شبیه سازهای موجود اغلب به دلیل معماری یکپارچه، فقدان مجموعه داده واقعی و محدوده کوچک برای به روزرسانی دوره ای، از ویژگی های مدیریت خدمات پیشرفته پشتیبانی نمی‌کنند.

شبیه ساز IfogSim، که یک شبیه ساز پرکاربرد که در محیط CloudSim توسعه داده شده است. شبیه ساز IfogSim یک کتابخانه با زبان برنامه نویسی جاوا می باشد. پروژه های مختلف مدیریت منابع در محیط‌های IfogSim، در آن قابل پیاده سازی است. از جمله مزایای این شبیه ساز می‌توان به موارد زیر اشاره نمود:

* در نظر گرفتن تأثیرات زمان
* این شبیه ساز کاملاً منعطف و کاربردی است.
* تست سیاست های مورد نظر در یک محیط قابل کنترل و قابل تکرار.
* تنظیم گلوگاه های سیستم قبل از استقرار روی ابر واقعی.
* پیکربندی قراردادهای SLA و چک کردن میزان تخطی روش های مختلف از آن.
* ارزیابی سیاست‌هایی مانند بهینه سازی روی سرویس دهنده های مورد استفاده.

با استفاده از جعبه ابزار IfogSim، بسیاری از عملیات مربوط به محاسبات ابری و لبه را می‌توان، شبیه سازی کرد. برخی از کاربردهای مهم این ابزار به شرح زیر می باشند:

* مدل سازی و شبیه سازی مراکز داده
* شبیه سازی ماشین های مجازی مبتنی بر مه همراه با تخصیص سیاست های مختلف بر روی آن ها
* درج پویای عناصر شبیه سازی، متوقف کردن و راه اندازی مجدد یک شبیه سازی
* شبیه سازی و مدل سازی توپولوژی های شبکه ای استفاده شده در مراکز مدیریت داده
* مدل سازی محاسبات مه و لبه
* نمایش چگونگی ساخت یک فضای مه و لبه
* چگونگی ساخت یک فضای مه و لبه
* چگونگی ساخت و پیکربندی محیط های مه و لبه
* نمایش گردش کار در مراکز داده ها و اختصاص دادن ساختار لبه به ماشین های مجازی و....

# 3-4) متغیرهای شبیه سازی

همان طور که بیان شد، برای ارزیابی و بررسی روش پیشنهادی و مقایسه آن با سایر رویکردها، از میحط شبیه سازی IfogSim و ویرایش گر NetBeans با زبان برنامه نویسی مبتنی بر جاوا و ماشین مجازی جاوا JDK استفاده‌شده است. در این ارزیابی ها، از یک کامپیوتر با سیستم عامل ویندوز 10 شرکت مایکروسافت نسخه 64 بیتی با پردازنده اینتل 4 هسته‌ای و رم پردازشی 8 گیگابایتی استفاده شد. جدول 4-1 نشان دهنده پارامترهای شبیه سازی برای روش پیشنهادی است. مجموعه داده مورد استفاده در پژوهش، مجموعه داده IfogSim می باشد که به صورت عمومی در اختیار محققان قرار داده شده است.

**جدول 4- 1) متغیرهای شبیه سازی**

|  |  |
| --- | --- |
| پارامترهای ارزیابی | مقادیر پارامتر |
| تعداد سرویس ها | 500 |
| اندازه بسته ورودی – خروجی | 50 کلیوبایت |
| بیشترین توان ارتباطی سرویس ها | 50 متر |
| توان ارتباطی معمول در سرویس ها | 20 متر |
| زمان مورد نیاز برای ارتباط | 3 تا 30 ثانیه برای هر ارتباط |
| بیشترین سطح ارتباطی | 50 ارتباط |
| پردازنه مرکزی (MIPS) | 200 تا 500 |
| مجموعه داده استفاده شده | مجموعه داده fog-computing-edge-servers |
| اندازه حافظه رم | 8 گیگابایت |
| حداکثر حجم داده | 900 مگابایت |
| زمان بند ابری | RoundRobin |
| تعداد کلودلت | 150 |
| تعداد هسته ها در هر ماشین مجازی | 2 هسته |

در این پژوهش از مجموعه داده ای استفاده می شود که به صورت آنلاین در اختیار محققان قرار داده شده است. این مخزن مجموعه ای از مجموعه داده های EUA را که محققان از منابع داده های دنیای واقعی جمع آوری کرده و نگهداری می‌کنند.

مجموعه داده ها به صورت عمومی منتشر شده است. ساختار کلی مجموعه داده به شرح زیر است:

* سرورهای ابر: شامل مجموعه داده های مکان های سرور ابر و پارامترهای کیفیت سرویس سرورها است.
* کاربران: شامل مجموعه داده های کاربر در محیط ابری است.
* درخواست ها: شامل وظایف درخواست شده از طریق کاربران برای پردازش در محیط ابر است.

همان طور که بیان شد، در این بخش از پژوهش، چارچوب شبیه سازی توسعه داده شده و الگوریتم پیشنهادی در محیط IfogSim ارزیابی شد. در این پژوهش از سناریوهای متفاوت با پارامترهای متفاوت برای شبیه سازی روش‌های مرتبط استفاده شد. یکی از مواردی که برای شبیه سازی استفاده شد کنترلر مبتنی بر محاسبات ابری با استفاده از ابزار IfogSimslab بود که یک ابزار مبتنی بر سرویس در محاسبات ابری است و به صورت کتابخانه به ابزار IfogSim اضافه می شود و پارامترهای کنترلی شبکه را به طور منظم از گره های شبکه جمع می‌کند. سپس، مکانیزم ماشین حالت برای تقلید از اقدامات کنترل کننده مبتنی بر مدل پیشنهادی به منظور مدیریت مصرف انرژی استفاده می شود، که شامل مولفه هایی مانند پارامترهای تعادل بار منابع یا افزایش توان انتقال اطلاعات است. چارچوب ارزیابی عملکرد و تنظیمات اصلی شبیه سازی در جدول 4-1 خلاصه شده‌است.

# 4-4) معیارهای عملکرد در فرایند شبیه سازی

برای ارزیابی عملکرد روش پیشنهادی برای زمانبندی در محیط‌های محاسباتی لبه و مه مبتنی بر یادگیری عمیق با استفاده از شبکه عصبی حافظه طولانی کوتاه مدت برای بهینه سازی بار سیستم و زمان پاسخ، پارامترها و معیارهای زیر درنظر گرفته می شوند.

* **میانگین تأخیر زمان:** میانگین تأخیر زمان بیانگر زمان مورد نیاز برای انتقال داده در پردازنده است.
* **درجه تفاوت سرویس دهنده جهت انتقال داده**: این متریک عملکرد برای اندازه گیری تعادل سرویس تخصیص داده شده استفاده می شود. تعادل مصرف انرژی و توازن بار ارتباط نزدیکی با این متریک عملکرد دارد.
* **کیفیت سرویس و زمان بهینه:** ویژگی خوب متریک عملکرد QoS است. این مؤلفه به صورت تعدادی از داده های ارتباطی است که با موفقیت در مقصد در زمان ارتباط موردنیاز دریافت می شود.
* **توان**: میزان کارایی، میزان تحویل پیام موفق از طریق کانال ارتباطی، در محیط محاسباتی مه است.
* **زمان بارگیری از سرور (DLTS):** زمان بارگیری از سرور در شبیه سازی ها اندازه گیری می شود تا عملکرد تأخیر زمانی از سرور در چارچوب های مختلف محاسباتی را در محیط محاسباتی مه نشان دهد.

# 5-4) نتایج شبیه سازی

در این بخش از پژوهش نتایج مربوط به ارزیابی های ارائه شده توسط روش پیشنهادی و رویکردهای پیشنهادی گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] مورد بررسی قرار گرفته و در قالب نمودار مقایسه می شوند. اولین مورد ارزیابی شامل تأخیر متوسط برای روش پیشنهادی و رویکردهای پیشنهادی گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] مقایسه شده‌است.

به طور کلی، تأخیر متوسط برابر میانگین تأخیر زمانی برای بهترین عملکرد از نظر تأخیر زمانی و ساختار روش پیشنهادی نشان می دهد که میانگین تأخیر زمانی با افزایش مقدار داده برای همه روش ها افزایش می یابد. مقادیر مربوط به تأخیر برای روش پیشنهادی و گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] در شکل 4-1 نشان داده شده است. این شکل نشان می دهد که با افزایش مقدار داده های ارتباطی برای هر دو، میانگین تأخیر زمان افزایش می‌یابد. اما افزایش تأخیر برای روش پیشنهادی نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] سطح پایین تری دارد، با توجه به اینکه روش پیشنهادی قبل از انجام محاسبات روی ساختار خود، از یک فرایند یادگیری عمیق به منظور پیش بینی سطح ترافیک و پرهیز از ایجاد ازدحام در سطح شبکه استفاده می کند، این ساختار موجب بهبود تعادل بار در شبکه شده و میانگین زمان انتظار در شبکه را کاهش می دهد. در نتیجه میانگین تأخیر برای روش پیشنهادی نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] سطح بهتری خواهد داشت.

**شکل 4- 1) مقایسه تأخیر میانگین روش پیشهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2]**

عملکرد سربار محاسباتی وظایف به روش های مختلف و برای تعداد متفاوتی از گره ها و مراکز داده هایی که توسط ساختار مدل پیشنهادی ارزیابی می شوند، در شکل 4-2 ارائه شده است. به طور کلی، نرخ سربار محاسباتی از هر استراتژی با افزایش مهلت و تعداد مراکز داده افزایش می یابد. با توجه به ارزیابی ها می‌توان دو مسئله را در نظر گرفت، مورد اول ارسال و تحویل اطلاعات بدون در نظر گرفتن مهلت زمانی پردازنده ها و مورد دوم با در نظر گرفتن مهلت زمانی است، بر اساس شکل 4-2 (الف) و بدون در نظر گرفتن مهلت زمانی برای پردازش، ساختار پیشنهادی، نرخ سربار محاسباتی کمتری در مقایسه با رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] دارد. با کاهش سربار محاسباتی، سرعت اجرای فرایند زمانبندی بهینه تر می شود. باتوجه به اینکه مهلت زمانی تأثیر بالایی بر سربار محاسباتی منابع دارد، به نظر می‌رسد با افزایش مهلت زمانی، این مقدار در روش پیشنهادی و رویکردهای مقایسه شده نیز سطح بالاتری به خود اختصاص داده و حجم پردازش‌های قابل اجرا در شبکه افزایش پیدا می‌کند.

**شکل 4- 2) مقایسه سربار محاسباتی روش پیشهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2]**

رویکرد پیشنهادی سربار محاسباتی بهتری نشان می دهد و در عین حال، رویکردهای مقایسه شده نیز سربار محاسباتی بالاتری ایجاد می‌کنند. با توجه به اینکه روش پیشنهادی از اطلاعات پالایش شده و پیش پردازش شده به منظور اجرای فرایند زمانبندی استفاده می‌کند، به نظر می‌رسد، در هر دو حالت استفاده از مهلت زمانی و عدم وجود مهلت زمانی، کارایی قابل توجهی نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] با محاسبه مه و لبه کارایی بالاتری را نشان می دهد.

در ادامه فرایند شبیه سازی، مقادیر مربوط به عملکرد روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] در زمینه تعادل بار ارزیابی می شود. تعادل بار در زمانبدی وظایف، یک متریک عملکرد برای شبکه بوده و شاخصی مفید برای ارزیابی کارایی است. این فرایند نشان می دهد که تعادل توزیع داده های دریافت شده از منبع تا چه اندازه موفق است. نتایج ارزیابی عملکرد تعادل بار برای روش پیشنهادی رویکرد مقایسه شده در شکل 4-3 نشان داده شده است.

**شکل 4- 3) مقایسه تعادل بار روش پیشهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2]**

بدیهی است، وقتی حجم داده به حد متفاوتی می‌رسد، مقادیر تعادل بار نیز متفاوت خواهد بود. با توجه به اینکه روش پیشنهادی مؤثرترین منبع برای زمانبندی را انتخاب خواهد کرد، در نتیجه مقادیر مربوط به عملکرد نرخ تعادل بار به دلیل توزیع بهینه منابع به وظایف نیز برای روش پیشنهادی بهتر از رویکردهای مقایسه شده خواهد بود. با توجه به اینکه در روش پیشنهادی از فرایند یادگیری عمیق برای بهبود ترافیک شبکه و از رویکرد محاسبات ابری برای توزیع بار ترافیکی شبکه استفاده می شود، لذا فرایند انتخاب منابع بهینه برای تخصیص درخواست در شبکه انجام شده و نتایج به صورت قابل توجهی بهبود پیدا می‌کنند. با بهبود تعادل بار، عملکرد کیفیت سرویس نیز نسبت به رویکرد مقایسه شده نیز بهبود پیدا می‌کند. در ادامه فرایند شبیه سازی ها، مقادیر مربوط به انرژی مصرفی برای رویکرد پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] با محاسبات ابری مورد بررسی و ارزیابی قرار می‌گیرد. انرژی مصرفی در مقادیر داده های مختلف برای روش پیشنهادی و رویکرد مقایسه شده در شکل 4-4 برای 5 مقیاس متفاوت از داده ها ارائه شده است. نتایج ارزیابی نشان می دهد مصرف انرژی برای روش پیشنهادی نسبت به رویکرد مقایسه شده گروه تحقیقاتی هوسین و همکارانش[21] سطح بهتری دارد ولی گروه تحقیقاتی وانگ و همکارانش[2] نسبت به روش پیشنهادی سطح پایین تری دارد.

**شکل 4- 4) مقایسه انرژی مصرفی روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2]**

با توجه به اینکه روش پیشنهادی از ساختار ارزیابی منابع مبتنی بر یادگیری ماشین در داخل شبکه استفاده می‌کند، خروجی حاصل از این فرایند، یک ساختار بهینه شده از نظر ترافیک شبکه می‌باشد. همچنین با استفاده از محاسبات ابری، روش پیشنهادی سعی در ایجاد تعادل بار ترافیکی در شبکه داشته و موجب افزایش پهنای باند قابل استفاده در شبکه می‌گردد، این فرایند به نوبه خود تأثیر بسزایی در انرژی مصرفی سیستم دارد. بر اساس شکل 4-4 مشاهده می شود که انرژی مصرفی برای روش پیشنهادی در تمامی حجم داده‌ای ارزیابی‌شده، نسبت به رویکرد مقایسه شده گروه تحقیقاتی وانگ و همکارانش[2] سطح بهتری دارد. در ادامه مقادیر مربوط به نرخ همگرایی به جواب بهینه برای روش پیشنهادی رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] مورد بررسی و ارزیابی قرار گرفته و نتایج ارزیابی در قالب نمودار در شکل 4-5 ارائه می شود.

**شکل 4- 5) مقایسه نرخ همگرایی به جواب بهینه برای روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2]**

نتایج ارزیابی نشان‌می دهد که روش پیشنهادی با زمانبدی وظایف بهینه در شبکه همگرایی بهتری به جواب بهینه دارد. همچنین با توجه به اینکه مسئله مدیریت ترافیک در روش پیشنهادی با استفاده از راهکار مبتنی بر یادگیری ماشین بهینه می شود. می توان توجیه کرد که راهکار پیشنهادی قادر به کمینه سازی تأخیر و در نهایت بهبود نرخ همگرایی به جواب بهینه است. در نهایت مقادیر مربوط به زمان پاسخ برای روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] در شکل 4-6 مورد بررسی قرار گرفته و نتایج نشان دهنده کارایی بالاتر روش پیشنهادی نسبت به رویکردهای مقایسه شده در مولفه زمان پاسخ است. با توجه به اینکه روش پیشنهادی با ایجاد تعادل بار در توزیع منابع پردازشی، باعث بهبود زمان اجرا و نرخ همگرایی به جواب بهینه می شود، از این رو می‌توان انتظار داشت که زمان پاسخ برای روش پیشنهادی به منظور اجرای وظایف پردازشی در مقایسه با رویکردهای گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] کارایی بالاتری داشته‌باشد.

**شکل 4- 6) مقایسه زمان پاسخ برای روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2]**

# 6-4) ارزیابی نتایج

در این فصل از پژوهش، رویکرد جدیدی در زمینه زمانبندی در محیط‌های محاسباتی لبه و مه مبتنی بر یادگیری عمیق با استفاده از شبکه عصبی حافظه طولانی کوتاه مدت برای بهینه سازی بار سیستم و زمان پاسخ ارائه و در محیط شبیه سازی IfogSim ارزیابی‌شده و نتایج حاصل از شبیه سازی، به صورت نمودار ارائه شدند. بر اساس نتایج به دست آمده از شبیه سازی مشخص شد که روش پیشنهادی نسبت به رویکرد مقایسه شده دارای تأخیر میانگین و زمان پاسخ بهتری است، وجود تعادل بار در روش پیشنهادی عاملی بر بهینگی روش پیشنهادی در تأخیر میانگین است. همچنین نتایج ارزیابی نشان داد که روش پیشنهادی تعادل بار بیشتری داشته و در نهایت قادر به تخصیص منابع بیشتر و در نتیجه تعادل بار بهتر در زمانبندی است. همچنین مقادیر مربوط به عملکرد بار محاسباتی برای روش پیشنهادی و رویکردهای مقایسه شده ارزیابی شدند و نتایج نشان دهنده کارایی بالای روش پیشنهادی نسبت به راهکارهای مقایسه شده بود. در ادامه نرخ همگرایی به جواب بهینه هر سه روش بر اساس مقادیر داده متفاوت ارزیابی شد و نتایج نشان داده که روش پیشنهادی به دلیل ارزیابی مبتنی بر ترافیک و تعادل بار بهینه، کارایی بالاتری نسبت به رویکرد مقایسه شده دارد. همچنین در فرایند شبیه سازی با ارزیابی انرژی مصرفی سرور مشاهده شد که روش پیشنهادی نست به رویکرد مقایسه شده، انرژی کمتری صرف تخصیص منابع مورد نظر دارد و در نتیجه کارایی روش پیشنهادی نسبت به رویکردهای مقایسه شده سطح بالاتری خواهد داشت.

5) فصل پنجم

# جمع‌بندی و پیشنهادات

**اهداف فصل:**

* ارائه جمع‌بندی کلی
* پیشنهادات آتی

# 1-5) جمع‌بندی

با توجه به پذیرش گسترده مجازی سازی و فناوری های ابری، امروزه شرکت ها و کاربران نهایی می‌توانند منابع محاسباتی را بر اساس تقاضا اجاره کنند. ظهور محاسبات ابری منجر به حضور روز افزون دستگاه های محاسباتی شبکه ای در فضاهای عمومی، تجاری و خصوصی می شود. ساختارهای محاسبات ابری می توانند قابلیت های محاسباتی و ذخیره سازی خود را در معرض نمایش بگذارند. تکثیر فناوری های محاسبات ابری در مقیاس کوچک و بزرگ برای حوزه های مختلف، مانند مراقبت های بهداشتی هوشمند، شهرهای هوشمند، شبکه های انرژی هوشمند، یا کارخانه های هوشمند امکان پذیر می‌سازد. با این حال، از نقطه نظر فناوری، ماهیت غیرمتمرکز محاسبات ابری با ساختار نسبتاً متمرکز ابر مطابقت ندارد. امروزه داده های محاسبات ابری عمدتاً به روش توزیع شده تولید می شوند، برای پردازش به یک ابر متمرکز ارسال می شوند و سپس به سایر دستگاه های محاسبات ابری، که اغلب نزدیک به منابع داده اولیه قرار دارند، تحویل داده می شوند.

این رویکرد پردازش متمرکز منجر به تأخیرهای ارتباطی بالا و نرخ انتقال داده پایین در محاسبات ابری و همچنین ساختارهای مجازی و کاربران بالقوه می شود. پشتیبانی از پردازش غیرمتمرکز داده ها در دستگاه های محاسبات ابری در ترکیب با مزایای فناوری های ابری و مجازی سازی به عنوان یک رویکرد امیدوارکننده برای کاهش هزینه های ارتباطی و زمان های انتقال داده در محاسبات ابری شناسایی شده است. برای تحقق پردازش غیر متمرکز داده، لازم است بخش هایی از منابع محاسباتی و ذخیره سازی مورد نیاز برای پردازش داده های موجود به منابع داده و مصرف کنندگان خدمات (یعنی کاربران نهایی یا مخزن های داده) نزدیک تر شوند. رویکرد مفهومی زیربنایی، یعنی مجازی سازی محاسبات ابری و استفاده از منابع مجازی برای پردازش داده ها، به عنوان محاسبات ابری شناخته می شود. جامعه هنوز در برابر تعاریف واضح این اصطلاحات همگرا نشده است. در محاسبات ابری، جایی که تعداد زیادی از دستگاه های شبکه‌ای ناهمگن با هم همکاری می‌کنند. این فرایند منجر به تأخیرهای ارتباطی کمتر و همچنین استفاده بهتر از منابع محاسباتی، ذخیره سازی و شبکه در حال حاضر موجود در محاسبات ابری می شود. موارد استفاده بالقوه برای محاسبات ابری شامل سناریوهای معمولی ارتباطی، به عنوان مثال، پیش فیلتر کردن داده ها در سناریوهای کلان داده، پیش پردازش جریان های داده از گره های حسگر، یا پردازش داده ها در سیستم های هوشمند است. در بسیاری از حوزه های کاربردی، محاسبات ابری در ترکیب با محاسبات مه و لبه استفاده می شود، که می‌تواند بر محدودیت در دسترس بودن منابع در مه با منابعی که بر حسب تقاضا به دست می آیند غلبه کند. در حالی که ایده اولیه و مبانی نظری محاسبات ابری در حال حاضر ایجاد شده است، هنوز کمبود راه حل های مشخص در مورد تأمین منابع وجود دارد. جدا از این سؤال که چگونه می توان منابع ارائه شده توسط محاسبات ابری را مجازی سازی کرد، یکی دیگر از موانع اصلی برای جذب محاسبات ابری، این سؤال است که چگونه خدمات محاسبات ابری را بر روی منابع مه موجود توزیع کرد.

همچنین از سویی دیگر، با توجه به ظهور محاسبات ابری خدمات فراوانی پدید آمده است و تداوم بخشیدن به همه این خدمات با استفاده از الگوی محاسبات ابری با پیچیدگی و مسائل بالایی مواجه شده است. به منظور مدیریت این مسئله، یک ارائه دهنده خدمات جدید به نام ترکیب سرویس محاسبات مه و لبه ارائه شده است تا خدمات را به طور مؤثر به کاربران نهایی ارائه دهد. عوامل قابل توجه متفاوتی مانند زمان پاسخگویی سرویس و کیفیت سرویس مورد انتظار وجود دارد که باید بدون نقض سایر محدودیت‌های منابع برآورده شود. در این راستا باید یک میان افزار به نام میان افزار کنترل ابری و یک معماری مبتنی تخصیص سرویس که فرایند تخصیص سرویس را با توجه به برخی محدودیت ها اجرا کند نیاز است. با استفاده از این معماری می‌توان مزایای نسل بعدی سیستم های کامپیوتری را به حداکثر رساند، اما این معماری نیازمند استراتژی های جدیدی برای مدیریت زمانبدی وظایف به منابع است. بدون شک مسئله زمانبدی وظایف، یک مسئله بهینه سازی ترکیبی است. راه حل بهینه به دست‌آمده، بار را با قرار دادن مناسب سرویس، فرایند اجرایی را به تعدادی از گره های مه توزیع می‌کند که منجر به کاهش ترافیک، بهبود تأخیر و افزایش تعادل بار می شود.

استفاده از چارچوب های ابری باعث بهبود ارتباطات، بهره وری، توسعه مقیاس کاری و صرفه جویی در زمان و هزینه برای شرکت ها می شود. این ساختارها همچنین می‌توانند به تعامل بین افراد و تجهیزات منجر شوند. طراحی مبتنی بر ابر همچنین می تواند به شرکت ها اجازه دهد تا از سیستم ها و تجهیزات بدون موانع فنی و اقتصادی بهره مند شوند. با توجه به مزایایی که محیط های محاسباتی مه و لبه می تواند برای صنایع و شرکت ها داشته باشد، این چارچوب می‌تواند بسیار جالب باشد. یکی از چالش‌های اصلی در این چارچوب های محاسباتی ابر، مشکل مکان سرویس است. بنابراین، در این پژوهش یک رویکرد زمانبندی وظایف در محیط محاسبات مه و لبه ارائه شد که می‌تواند به عنوان یک راه حل عملی توسط سازمان های فعال در زمینه پردازش ابر و مه استفاده شود. بهبود فرآیندهای زمانبندی با تشریح پارامترهای کیفیت خدمات مورد استفاده در فرآیند تخصیص و مدیریت منابع با استفاده از روش پیشنهادی شرح داده شد.

در این پژوهش رویکرد پیشنهادی در زمینه زمانبندی در محیط های محاسباتی لبه و مه مبتنی بر یادگیری عمیق با استفاده از شبکه عصبی حافظه طولانی کوتاه مدت برای بهینه سازی بار سیستم و زمان پاسخ در محیط شبیه سازی IfogSim پیاده سازی شد و نتایج شبیه سازی در قالب نمودار از نتایج به دست‌آمده از شبیه سازی ارائه شد. مشخص شد که روش پیشنهادی موجب بهبود 4.52 درصدی و 3.53 درصدی تأخیر در شبکه نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] ارائه کرده است. وجود متعادل کننده بار در روش پیشنهادی یکی از عوامل مناسب بودن روش پیشنهادی در تأخیر متوسط است، همچنین نتایج ارزیابی نشان می دهد که روش پیشنهادی تعادل بار بالایی برخوردار است و به اندازه 2.63 درصد و 3.21 درصد بهبود نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] شده را نشان می دهد. همچنین نتایج نشان دهنده کارایی بهتر نرخ سربار محاسباتی منابع برای روش پیشنهادی به اندازه 2.11 درصد و 1.89 درصد نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] مقایسه شده نشان می دهد. مقایسه مصرف انرژی برای روش پیشنهادی رویکرد گروه تحقیقاتی وانگ و همکارانش[2] نیز به طور میانگین بهبود 1.98 درصدی داشت اما نسبت به گروه تحقیقاتی هوسین و همکارانش[21] 2.32 درصد روش پیشنهادی عملکرد ضعیف تر را نشان می دهد. همچنین نتایج ارزیابی نشان دهنده نرخ همگرایی بهتر برای روش پیشنهادی به اندازه 1.98 درصد و 2.53 درصد نسبت به رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] را نشان می دهد. در نهایت مقایسه زمان پاسخ برای روش پیشنهادی و رویکرد گروه تحقیقاتی هوسین و همکارانش[21] و گروه تحقیقاتی وانگ و همکارانش[2] نشان دهنده بهبود 2.21 و 1.93 درصدی است.

# 2-5) پیشنهادات آتی

عنوان پیشنهاد برای کارهای آتی می‌توان به موارد زیر اشاره نمود.

* به منظور بالا بردن فرایند زمانبندی وظایف می‌توان از روش های چند معیاره با ایجاد وزن بر روی پارامترهای مربوط به انتخاب گره استفاده کرد.
* استفاده از رویکردهای بهینه سازی ریاضی و همچنین رویکردهای ترکیبی جهت زمانبندی وظایف در محیط ابر
* ارزیابی و بررسی کارایی راهکار پیشنهادی در مقیاس های بزرگ تر و واقعی تر
* ارزیابی روش پیشنهادی براساس پارامترهای متفاوت دیگر از کیفیت سرویس منابع پردازشی
* استفاده از رویکردهای متفاوت بهینه سازی مصرف انرژی و کاهش هزینه مدیریت منابع پردازشی

# منابع و مراجع

[1] H. S. Ali, R. R. Rout, P. Parimi, and S. K. Das, “Real-Time Task Scheduling in Fog-Cloud Computing Framework for IoT Applications: A Fuzzy Logic based Approach, ” in *2021 International Conference on COMmunication Systems NETworkS (COMSNETS)*, Jan. 2021, pp. 556–564. doi: 10.1109/COMSNETS51098.2021.9352931.

[2] Z. Wang, M. Goudarzi, M. Gong, and R. Buyya, “Deep Reinforcement Learning-based Scheduling for Optimizing System Load and Response Time in Edge and Fog Computing Environments, ” Oct. 22, 2023, *arXiv*: arXiv:2309.07407. doi: 10.48550/arXiv.2309.07407.

[3] F. Hoseiny, S. Azizi, M. Shojafar, F. Ahmadiazar, and R. Tafazolli, “PGA: A Priority-aware Genetic Algorithm for Task Scheduling in Heterogeneous Fog-Cloud Computing, ” in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2021, pp. 1–6. doi: 10.1109/INFOCOMWKSHPS51825.2021.9484436.

[4] S. Ijaz, E. U. Munir, S. G. Ahmad, M. M. Rafique, and O. F. Rana, “Energy-makespan optimization of workflow scheduling in fog–cloud computing, ” *Computing*, vol. 103, no. 9, pp. 2033–2059, Sep. 2021, doi: 10.1007 / s00607-021-00930-0.

[5] S.-E. Chafi, Y. Balboul, M. Fattah, S. Mazer, and M. E. Bekkali, “Enhancing IoT Workloads: A Comparative Study of Fog and Edge Computing Scheduling Algorithms,” in *2024 International Conference on Circuit, Systems and Communication (ICCSC)*, Jun. 2024, pp. 1–6. doi: 10.1109/ICCSC62074.2024.10616815.

[6] M. Abdel-Basset, R. Mohamed, M. Elhoseny, A. K. Bashir, A. Jolfaei, and N. Kumar, “Energy-Aware Marine Predators Algorithm for Task Scheduling in IoT-Based Fog Computing Applications, ” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 5068–5076, Jul. 2021, doi: 10.1109/TII.2020.3001067.

[7] N. Alshammari, S. S. Gill, H. Pervaiz, Q. Ni, and H. Ahmed, “Resource Scheduling in Integrated IoT and Fog Computing Environments: A Taxonomy, Survey and Future Directions, ” in *Resource Management in Distributed Systems*, A. Mukherjee, D. De, and R. Buyya, Eds., Singapore: Springer Nature, 2024, pp. 63–77. doi: 10.1007/978-981-97-2644-8\_4.

[8] M. Jia, J. Zhu, and H. Huang, “Energy and delay-ware massive task scheduling in fog-cloud computing system, ” *Peer-to-Peer Netw. Appl.*, vol. 14, no. 4, pp. 2139–2155, Jul. 2021, doi: 10.1007/s12083-021-01118-1.

[9] M. Abdul Nabi and I. H. El-adaway, “Understanding the Key Risks Affecting Cost and Schedule Performance of Modular Construction Projects, ” *Journal of Management in Engineering*, vol. 37, no. 4, p. 04021023, Jul. 2021, doi: 10.1061/(ASCE)ME.1943-5479.0000917.

[10] R. Ghafari and N. Mansouri, “A novel energy-based task scheduling in fog computing environment: an improved artificial rabbits optimization approach, ” *Cluster Comput*, vol. 27, no. 6, pp. 8413–8458, Sep. 2024, doi: 10.1007/s10586-024-04396-5.

[11] P. Memari, S. S. Mohammadi, F. Jolai, and R. Tavakkoli-Moghaddam, “A latency-aware task scheduling algorithm for allocating virtual machines in a cost-effective and time-sensitive fog-cloud architecture, ” *J Supercomput*, vol. 78, no. 1, pp. 93–122, Jan. 2022, doi: 10.1007/s11227-021-03868-4.

[12] R. O. Aburukba, T. Landolsi, and D. Omer, “A heuristic scheduling approach for fog-cloud computing environment with stationary IoT devices, ” *Journal of Network and Computer Applications*, vol. 180, p. 102994, Apr. 2021, doi: 10.1016/j.jnca.2021.102994.

[13] H. F. Atlam, R. J. Walters, and G. B. Wills, “Fog Computing and the Internet of Things: A Review, ” *Big Data and Cognitive Computing*, vol. 2, no. 2, Art. no. 2, Jun. 2018, doi: 10.3390/bdcc2020010.

[14] M. Mokni, S. Yassa, J. E. Hajlaoui, R. Chelouah, and M. N. Omri, “Cooperative agents-based approach for workflow scheduling on fog-cloud computing, ” *J Ambient Intell Human Comput*, Apr. 2021, doi: 10.1007/s12652-021-03187-9.

[15] AR. Arunarani, D. Manjula, and V. Sugumaran, “Task scheduling techniques in cloud computing: A literature survey, ” *Future Generation Computer Systems*, vol. 91, pp. 407–415, Feb. 2019, doi: 10.1016/j.future.2018.09.014.

[16] T. S. Nikoui, A. Balador, A. M. Rahmani, and Z. Bakhshi, “Cost-Aware Task Scheduling in Fog-Cloud Environment, ” in *2020 CSI/CPSSI International Symposium on Real-Time and Embedded Systems and Technologies (RTEST)*, Jun. 2020, pp. 1–8. doi: 10.1109/RTEST49666.2020.9140118.

[17] P. Lai *et al.*, “QoE-aware user allocation in edge computing systems with dynamic QoS, ” *Future Generation Computer Systems*, vol. 112, pp. 684–694, Nov. 2020, doi: 10.1016/j.future.2020.06.029.

[18] D. Xu *et al.*, “Edge Intelligence: Architectures, Challenges, and Applications, ” Jun. 12, 2020, *arXiv*: arXiv:2003.12172. doi: 10.48550/arXiv.2003.12172.

[19] S. Davy *et al.*, “Challenges to support edge-as-a-service, ” *IEEE Communications Magazine*, vol. 52, no. 1, pp. 132–139, Jan. 2014, doi: 10.1109/MCOM.2014.6710075.

[20] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang, and W. Shi, “Edge Computing for Autonomous Driving: Opportunities and Challenges, ” *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1697–1716, Aug. 2019, doi: 10.1109/JPROC.2019.2915983.

[21] M. Hussain, S. Nabi, and M. Hussain, “RAPTS: resource aware prioritized task scheduling technique in heterogeneous fog computing environment, ” *Cluster Comput*, vol. 27, no. 9, pp. 13353–13377, Dec. 2024, doi: 10.1007/s10586-024-04612-2.

[22] F. Hoseiny, S. Azizi, M. Shojafar, and R. Tafazolli, “Joint QoS-aware and Cost-efficient Task Scheduling for Fog-cloud Resources in a Volunteer Computing System, ” *ACM Trans. Internet Technol.*, vol. 21, no. 4, p. 86:1-86:21, Jul. 2021, doi: 10.1145/3418501.

[23] A. Lakhan, Q.-U.-A. Mastoi, M. Elhoseny, M. S. Memon, and M. A. Mohammed, “Deep neural network-based application partitioning and scheduling for hospitals and medical enterprises using IoT assisted mobile fog cloud,” *Enterprise Information Systems*, vol. 0, no. 0, pp. 1–23, Feb. 2021, doi: 10.1080/17517575.2021.1883122.

[24] K. Dubey, S. C. Sharma, and M. Kumar, “A Secure IoT Applications Allocation Framework for Integrated Fog-Cloud Environment, ” *J Grid Computing*, vol. 20, no. 1, p. 5, Feb. 2022, doi: 10.1007/s10723-021-09591-x.

[25] J. Bisht and V. S. Vampugani, “Load and Cost-Aware Min-Min Workflow Scheduling Algorithm for Heterogeneous Resources in Fog, Cloud, and Edge Scenarios, ” *IJCAC*, vol. 12, no. 1, pp. 1–20, Jan. 2022, doi: 10.4018/IJCAC.2022010105.

[26] S. D. Okegbile, B. T. Maharaj, and A. S. Alfa, “A multi-user tasks offloading scheme for integrated edge-fog-cloud computing environments, ” *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2022, doi: 10.1109/TVT.2022.3167892.

[27] M. Mokni and S. Yassa, “MAS-aware Approach for QoS-based IoT Workflow Scheduling in Fog-Cloud Computing, ” in *Optimization and Machine Learning*, John Wiley & Sons, Ltd, 2022, pp. 25–53. doi: 10.1002/9781119902881.ch2.

[28] C. Jian, L. Bao, and M. Zhang, “A high-efficiency learning model for virtual machine placement in mobile edge computing, ” *Cluster Comput*, Jan. 2022, doi: 10.1007/s10586-022-03550-1.

[29] N. Sarrafzade, R. Entezari-Maleki, and L. Sousa, “A genetic-based approach for service placement in fog computing, ” *J Supercomput*, vol. 78, no. 8, pp. 10854–10875, May 2022, doi: 10.1007/s11227-021-04254-w.

[30] J. Taghizadeh, M. Ghobaei‐Arani, and A. Shahidinejad, “A metaheuristic‐based data replica placement approach for data‐intensive IoT applications in the fog computing environment, ” *Softw Pract Exp*, vol. 52, no. 2, pp. 482–505, Feb. 2022, doi: 10.1002/spe.3032.

[31] M. Ghobaei-Arani and A. Shahidinejad, “A cost-efficient IoT service placement approach using whale optimization algorithm in fog computing environment, ” *Expert Systems with Applications*, vol. 200, p. 117012, Aug. 2022, doi: 10.1016/j.eswa.2022.117012.

[32] M. Salimian, M. Ghobaei-Arani, and A. Shahidinejad, “An Evolutionary Multi-objective Optimization Technique to Deploy the IoT Services in Fog-enabled Networks: An Autonomous Approach, ” *Applied Artificial Intelligence*, vol. 36, no. 1, p. 2008149, Dec. 2022, doi: 10.1080/08839514.2021.2008149.

[33] M. Tekiyehband, M. Ghobaei-Arani, and A. Shahidinejad, “An efficient dynamic service provisioning mechanism in fog computing environment: A learning automata approach, ” *Expert Systems with Applications*, vol. 198, p. 116863, Jul. 2022, doi: 10.1016/j.eswa.2022.116863.

[34] T. Djemai, P. Stolf, T. Monteil, and J.-M. Pierson, “A Discrete Particle Swarm Optimization Approach for Energy-Efficient IoT Services Placement Over Fog Infrastructures,” in *2019 18th International Symposium on Parallel and Distributed Computing (ISPDC)*, Jun. 2019, pp. 32–40. doi: 10.1109/ISPDC.2019.00020.

[35] K. Velasquez, D. P. Abreu, L. Paquete, M. Curado, and E. Monteiro, “A Rank-based Mechanism for Service Placement in the Fog, ” in *2020 IFIP Networking Conference (Networking)*, Jun. 2020, pp. 64–72.

[36] F. Khosroabadi, F. Fotouhi-Ghazvini, and H. Fotouhi, “SCATTER: Service Placement in Real-Time Fog-Assisted IoT Networks, ” *Journal of Sensor and Actuator Networks*, vol. 10, no. 2, Art. no. 2, Jun. 2021, doi: 10.3390/jsan10020026.

[37] J. Choi and S. Ahn, “Scalable Service Placement in the Fog Computing Environment for the IoT-Based Smart City, ” *Journal of Information Processing Systems*, vol. 15, no. 2, pp. 440–448, 2019, doi: 10.3745/JIPS.03.0113.

[38] M. Ayoubi, M. Ramezanpour, and R. Khorsand, “An autonomous IoT service placement methodology in fog computing, ” *Software: Practice and Experience*, vol. 51, no. 5, pp. 1097–1120, 2021, doi: 10.1002/spe.2939.

[39] J. C. Guevara and N. L. S. da Fonseca, “Task scheduling in cloud-fog computing systems, ” *Peer-to-Peer Netw. Appl.*, vol. 14, no. 2, pp. 962–977, Mar. 2021, doi: 10.1007/s12083-020-01051-9.

[40] A. Najafizadeh, A. Salajegheh, A. M. Rahmani, and A. Sahafi, “Multi-objective Task Scheduling in cloud-fog computing using goal programming approach, ” *Cluster Comput*, vol. 25, no. 1, pp. 141–165, Feb. 2022, doi: 10.1007/s10586-021-03371-8.

[41] A. S. M. S. Hosen, P. K. Sharma, and G. H. Cho, “MSRM-IoT: A Reliable Resource Management for Cloud, Fog, and Mist-Assisted IoT Networks, ” *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2527–2537, Feb. 2022, doi: 10.1109/JIOT.2021.3090779.

[42] Y. Zhang, X. Zhang, S. Ning, J. Gao, and Y. Liu, “Energy-Efficient Multilevel Heterogeneous Routing Protocol for Wireless Sensor Networks, ” *IEEE Access*, vol. 7, pp. 55873–55884, 2019, doi: 10.1109/ACCESS.2019.2900742.

[43] J. Benesty, J. Chen, Y. Huang, and I. Cohen, “Pearson Correlation Coefficient, ” in *Noise Reduction in Speech Processing*, I. Cohen, Y. Huang, J. Chen, and J. Benesty, Eds., in Springer Topics in Signal Processing., Berlin, Heidelberg: Springer, 2009, pp. 1–4. doi: 10.1007/978-3-642-00296-0\_5.

**یوست الف) واژه‌نامه انگلیسی به فارسی**

الف- 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Appropriate | مناسب |  | Hierarchical | سلسله‌مراتبی |
| Attribute | ویژگی |  | Infrastructure | زیرساخت |
| Capability | قابلیت |  | Keyword | کلمه کلیدی |
| Centralized | متمرکز |  | Knowledge sharing | به اشتراک‌گذاری دانش |
| Challenge | چالش |  | Matching | انطباق |
| Distributed system | سیستم توزیع‌شده |  | Lexical | لغوی |
| Provider | فراهم‌کننده |  | Monitoring | نظارت |
| Common | مشترک |  | Neighbor | همسایه |
| Concept | مفهوم |  | Node | گره |
| Crowdsourcing | جمع‌سپاری |  | Numerical | عددی |
| Effective | مؤثر |  | Observation | مشاهده |
| Equivalent | هم‌ارز |  | Ontology | هستی‌شناسی |
| Evaluate | ارزیابی |  | Outsource | برون‌سپاری |
| Resource | منبع |  | Peer to peer | نظیر به نظیر |
| Fault Tolerance | تحمل‌پذیری خرابی |  | Query | پرس‌وجو |
| Feature | مشخصه |  | Reasoning | استدلال |
| Resource | منبع |  | Relevant | مرتبط |
| Semantic | معنایی |  | Similarity | شباهت |
| SDN | شبکه نرم‌افزار محور |  | Subset | زیرمجموعه |

**پیوست ب) واژه‌نامه فارسی به انگلیسی**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SDN | شبکه نرم‌افزار محور |  | Evaluate | ارزیابی |
| Keyword | کلمه کلیدی |  | Reasoning | استدلال |
| Node | گره |  | Matching | انطباق |
| Lexical | لغوی |  | Outsource | برون‌سپاری |
| Centralized | متمرکز |  | sharing | به اشتراک‌گذاری |
| Distributed system | سیستم توزیع‌شده |  | Query | پرس‌وجو |
| Observation | مشاهده |  | Fault Tolerance | تحمل‌پذیر خرابی |
| Common | مشترک |  | Topology | توپولوژی |
| Feature | مشخصه |  | Crowdsourcing | جمع سپاری |
| Semantic | معنایی |  | Challenge | چالش |
| Concept | مفهوم |  | Automatic | خودکار |
| Appropriate | مناسب |  | Understanding | درک |
| Resource | منبع |  | Infrastructure | زیرساخت |
| Expert Resource | منبع متخصص |  | Subset | زیرمجموعه |
| Effective | مؤثر |  | Hierarchical | سلسله‌مراتبی |
| Heterogeneous | ناهمگن |  | Similarity | شباهت |
| Monitoring | نظارت |  | Numerical | عددی |
| Peer to peer | نظیر به نظیر |  | Relevant | مرتبط |
|  |  |  |  |  |

الف-2

**Abstract**

**Keywords:**



**Salman Institute of Higher Education**

**Computer engineering department**

**Master's Thesis (M. Sc.)**

**in the field of information technology engineering - computer networks**

**Title:**

**Scheduling in edge and fog computing environments based on deep learning using long-short-term memory neural network to optimize system load and response time**

**Instructor:**

**Dr.**

**Writing:**

**Winter 1403**

1. IoT [↑](#footnote-ref-1)
2. Quality of Service [↑](#footnote-ref-2)
3. خیلی سخت [↑](#footnote-ref-3)
4. Round-Trip Time [↑](#footnote-ref-4)
5. Pearson's Correlation Coefficient [↑](#footnote-ref-5)
6. Correlation Coefficient [↑](#footnote-ref-6)
7. Multi Layer Perceptron [↑](#footnote-ref-7)
8. Forget Gate [↑](#footnote-ref-8)
9. Sigmoid function [↑](#footnote-ref-9)
10. Input Gate [↑](#footnote-ref-10)
11. Output Gate [↑](#footnote-ref-11)