New Researches in The Smart City
Vol 4 (Issue 3), 6-21
nrsc.apadana.ac.ir

Apadana Institute of Higher Education

Efficiency of Task Scheduling in Edge Computing for IoT and Blockchain Using Metaheuristic Algorithms

Ali Etedadi*

M.Sc. in Information Technology Engineering - Computer Networks, Islamic Azad University, Lahijan Branch, Guilan, Iran

Received: 30/December/2026 Accepted: 19/February/2026 eISSN: 3060-6144 ISSN: 2980-8936

Abstract

With the rapid expansion of the Internet of Things, fifth-generation networks, and intelligent systems, a massive volume of real-time data is being generated, the processing of which requires low latency, high efficiency, and optimal resource management. In this context, edge computing has emerged as an effective solution to transfer part of the processing from the cloud to the vicinity of the data source. However, the limited computational and energy resources of edge nodes have made task scheduling one of the fundamental challenges of this architecture. This study investigates the efficiency of scheduling methods based on metaheuristic algorithms in edge computing environments, with a focus on the Internet of Things and the capability of integration with blockchain. To this end, a three-layer model comprising IoT devices, edge nodes, and cloud infrastructure is presented, and the task allocation problem is modeled as a multi-objective optimization problem. In the proposed method, genetic algorithm and particle swarm optimization are used to reduce latency, improve quality of service, and increase resource utilization. Simulation results showed that the genetic algorithm, with an average latency of 65 milliseconds and resource utilization of 88%, performs better than Greedy, Round Robin, and Random methods. The findings indicate that the use of metaheuristic algorithms can play an effective role in optimizing task scheduling and improving the efficiency of IoT–Edge systems.

Keywords: edge computing, internet of things, blockchain, metaheuristic algorithms, fog computing

* Corresponding Author: etedadii.alii@gmail.com

فصلنامه پژوهش‌های نوین در شهر هوشمند
دوره ۴، شماره ۳، بهار ۱۴۰۵، ۶-۲۱
nrsc.apadana.ac.ir

مؤسسه آموزش عالی آپادانا
غیر دولتی - غیر انتفاعی

تاریخ دریافت: ۱۴۰۵/۰۲/۰۹ تاریخ پذیرش: ۱۴۰۵/۰۳/۳۰ eISSN: 3060-6144 ISSN: 2980-8936

کارآمدی روش زمان‌بندی در پردازش لبه با نگاه به اینترنت اشیا و بلاکچین با استفاده از الگوریتم‌های فراابتکاری

علی اعتدادی*

کارشناسی‌ارشد مهندسی فناوری اطلاعات- شبکه‌های کامپیوتری، دانشگاه آزاد اسلامی واحد لاهیجان، گیلان، ایران

چکیده

با گسترش سریع اینترنت اشیا، شبکه‌های نسل پنجم و سامانه‌های هوشمند، حجم عظیمی از داده‌های بلادرنگ تولید می‌شوند که پردازش آن‌ها نیازمند تأخیر کم، بهره‌وری بالا و مدیریت بهینه منابع است. در این میان، پردازش لبه به‌عنوان راهکاری مؤثر برای انتقال بخشی از پردازش‌ها از ابر به نزدیکی منبع داده مطرح شده است. اما، محدودیت منابع محاسباتی و انرژی در گره‌های لبه، مسئله‌ی زمان‌بندی وظایف را به یکی از چالش‌های اساسی این معماری تبدیل کرده است. در پژوهش حاضر، کارآمدی روش‌های زمان‌بندی مبتنی بر الگوریتم‌های فراابتکاری در محیط‌های پردازش لبه با تمرکز بر اینترنت اشیا و قابلیت یکپارچه‌سازی با بلاکچین بررسی شده است. بدین منظور، یک مدل سه‌لایه شامل دستگاه‌های IoT، گره‌های لبه و زیرساخت ابری ارائه شده و مسئله‌ی تخصیص وظایف به‌صورت یک مسئله‌ی بهینه‌سازی چندهدفه مدل‌سازی گردیده است. در روش پیشنهادی، از الگوریتم ژنتیک و بهینه‌سازی ازدحام ذرات برای کاهش تأخیر، بهبود کیفیت خدمات و افزایش بهره‌وری منابع استفاده شده است. نتایج شبیه‌سازی نشان دادند که الگوریتم ژنتیک با میانگین تأخیر ۶۵ میلی‌ثانیه و بهره‌وری منابع ۸۸ درصد، عملکرد بهتری نسبت به روش‌های Greedy، Round Robin و Random دارد. لذا، یافته‌های پژوهش بیانگر آن هستند که استفاده از الگوریتم‌های فراابتکاری می‌تواند نقش مؤثری در بهینه‌سازی زمان‌بندی وظایف و ارتقای کارایی سامانه‌های IoT–Edge ایفا کند.

کلیدواژه‌ها: پردازش لبه، اینترنت اشیا، بلاکچین، الگوریتم‌های فراابتکاری، رایانش مه

* نویسندهٔ مسئول: etedadii.alii@gmail.com

مقدمه

اینترنت اشیا را می‌توان به‌عنوان شبکه‌ای گسترده از اشیای هوشمند در نظر گرفت که به انواع حسگرها و ماژول‌های ارتباطی مجهز هستند و قادرند داده‌های محیطی را جمع‌آوری، تبادل و تحلیل کنند. در چنین ساختاری، سامانه‌های کاربردی با بهره‌گیری از اطلاعات دریافت‌شده از حسگرها، تصمیمات لازم را برای کنترل و مدیریت اشیا اتخاذ می‌کنند تا نیازهای مرتبط با آن‌ها برآورده شوند. کاربردهایی نظیر کشاورزی هوشمند، واقعیت افزوده و واقعیت مجازی از جمله نمونه‌های مهم به‌کارگیری اینترنت اشیا محسوب می‌شوند. این فناوری به‌عنوان یکی از مگاترندهای تحول‌آفرین در صنعت شناخته می‌شود که نقش مهمی در شکل‌گیری نسل جدید صنایع هوشمند ایفا می‌کند (Aly et al., 2021). بر اساس پیش‌بینی‌ها، تعداد دستگاه‌های متصل به اکوسیستم اینترنت اشیا تا سال ۲۰۳۰ به حدود سی میلیارد دستگاه خواهد رسید (Alfawaz, 2022).

با گسترش فناوری‌هایی مانند اینترنت اشیا، 5G و سامانه‌های هوشمند، حجم عظیمی از داده‌های لحظه‌ای تولید می‌شوند. پردازش لبه، به‌عنوان راهکاری نوین برای پردازش این داده‌ها، در نزدیکی منبع تولید (به جای ابر) معرفی شده است که مزایایی مانند کاهش تأخیر، صرفه‌جویی در پهنای باند و کاهش وابستگی به ابر را به همراه دارد.

با این حال، چالش اصلی این معماری مصرف بالای انرژی در گره‌های لبه است؛ زیرا این گره‌ها منابع محاسباتی و انرژی محدودی دارند. افزایش مصرف انرژی می‌تواند طول عمر تجهیزات را کاهش دهد، هزینه‌ها را افزایش دهد و کیفیت خدمات[1] را تحت تأثیر قرار دهد.

از سوی دیگر، ماهیت پویا، توزیع‌شده و ناهمگن محیط‌های پردازش لبه سبب می‌شوند که روش‌های سنتی زمان‌بندی نتوانند به‌طور مؤثر پاسخ‌گوی پیچیدگی‌های این محیط‌ها باشند. به همین دلیل، در سال‌های اخیر استفاده از الگوریتم‌های فراابتکاری[2] به‌عنوان رویکردی کارآمد برای حل مسائل بهینه‌سازی پیچیده و چندهدفه مورد توجه قرار گرفته است. این الگوریتم‌ها که اغلب با الهام از رفتارهای طبیعی و اجتماعی موجودات زنده طراحی شده‌اند، توانایی جست‌وجو در فضاهای بزرگ و غیرخطی راه‌حل را داشته و می‌توانند در حضور محدودیت‌های متعدد، پاسخ‌های نزدیک به بهینه ارائه دهند (Dastjerdi et al., 2016; Mir et al., 2023).

برخی از مطالعات پیشین در حوزه الگوریتم‌های زمان‌بندی

به‌طور کلی، معماری سیستم‌های مورد استفاده در مسئله‌ی زمان‌بندی در محیط‌های اینترنت اشیا و پردازش لبه را می‌توان در دو دسته‌ی اصلی طبقه‌بندی کرد: دسته‌ی نخست شامل معماری‌های مبتنی بر مدل سه‌لایه است و دسته دوم، معماری‌هایی را در بر می‌گیرد که با در نظر گرفتن مدل چهارلایه به حل مسئله‌ی زمان‌بندی می‌پردازند (Karimi et al., 2022).

در مقاله (Dastjerdi et al., 2016) یک معماری سه‌لایه ارائه شده است که در آن با بهره‌گیری از چند الگوریتم زمان‌بندی، شامل تأخیر، اولویت و هم‌زمان، یک زمان‌بند آگاه به تحرک ارائه گردیده است. با این حال، یکی از چالش‌های مطرح در این پژوهش، هزینه‌ی محاسباتی بالا و کارایی محدود الگوریتم‌های قطعی در مواجهه با مسائلی با فضای حالت بزرگ عنوان شده است.

در مقاله (Bittencourt et al.2017) نیز یک معماری سه‌لایه معرفی شده که در آن یک الگوریتم زمان‌بندی آگاه به حرکت برای کاربردهای حساس به تأخیر در حوزه‌ی سلامت و مراقبت‌های پزشکی طراحی شده است. فرایند زمان‌بندی در این روش در دو مرحله انجام می‌شود؛ در مرحله‌ی نخست، وظایف بر اساس میزان اهمیت و فوریت

1. QoS
2. metaheuristic algorithms

رتبه‌بندی می‌شوند و در مرحله‌ی دوم، با استفاده از یک الگوریتم ابتکاری و با در نظر گرفتن الگوی حرکت کاربران، تخصیص وظایف به گره‌های پردازشی انجام می‌گیرد. با این حال، یکی از نکات قابل توجه در این پژوهش، عدم تفکیک دقیق فعالیت‌هایی است که در سطوح مختلف پردازش انجام می‌شوند؛ به‌عبارت دیگر، تخصیص تمامی انواع وظایف و فعالیت‌ها به سطح پردازش مه می‌تواند منجر به کاهش کارایی سیستم شود.

در مقاله (Abdelmoneem et al., 2020) نیز یک معماری سه‌لایه در نظر گرفته شده است که در آن گره‌های پردازش مه در لایه لبه قرار دارند. تمرکز اصلی این پژوهش بر کاهش مصرف انرژی در گره‌های پردازشی مه است و برای دستیابی به این هدف، از یک روش فراابتکاری استفاده شده است. با وجود مزایای مطرح‌شده، یکی از محدودیت‌های این پژوهش در نظر گرفتن تنها یک سرور مه در پیاده‌سازی مدل پیشنهادی است؛ در حالی‌که در محیط‌های واقعی، ارتباط و تعامل میان چندین سرور مه می‌تواند تأثیر قابل‌توجهی بر شاخص‌های ارزیابی عملکرد سیستم داشته باشد (Shahvaroughi Farahani et al., 2023).

همچنین، در مقاله (Huang et al.,2020) یک مدل معماری دولایه برای بهینه‌سازی مصرف انرژی از طریق زمان‌بندی در فرایند تخصیص وظایف به گره‌های پردازشی ارائه شده است. در این رویکرد، ارتباط میان دستگاه‌های اینترنت اشیا و گره‌های پردازشی مبنای انتخاب گره مناسب برای اجرای هر وظیفه قرار می‌گیرد؛ به این صورت که هر دستگاه مشخصات وظیفه‌ی خود را به گره‌های پردازشی در دسترس ارسال می‌کند و این گره‌ها پس از ارزیابی توان پردازشی خود، پاسخ مناسبی در خصوص امکان اجرای وظیفه به دستگاه ارسال می‌کنند. با این حال، یکی از محدودیت‌های مهم این مدل معماری، در نظر نگرفتن زیرساخت پردازش ابری به‌عنوان یک منبع پردازشی مکمل است، در حالی‌که در بسیاری از کاربردهای اینترنت اشیا، مانند کشاورزی هوشمند، همه وظایف نیازمند پردازش بلادرنگ نیستند و می‌توان با طبقه‌بندی مناسب وظایف، از ظرفیت‌های پردازشی در هر دو سطح لبه و ابر به‌صورت کارآمد بهره گرفت (Mohammadi & Ezadi Yegane, 2021).

مدل سیستمی

به‌منظور بهینه‌سازی مصرف انرژی در معماری پردازش لبه، در این پژوهش مدلی ساده، قابل پیاده‌سازی و در عین حال، کارآمد برای تخصیص و زمان‌بندی وظایف پردازشی ارائه می‌شود. ساختار سیستم پیشنهادی از سه بخش اصلی شامل دستگاه‌های اینترنت اشیا،[1] گره‌های لبه[2] و در صورت نیاز سرورهای ابری[3] تشکیل شده است. در این چارچوب، وظایف تولیدشده توسط دستگاه‌های اینترنت اشیا باید با حداقل تأخیر و کمترین میزان مصرف انرژی پردازش شوند. با توجه به محدودیت منابع محاسباتی و انرژی در گره‌های لبه، تصمیم‌گیری درباره‌ی محل اجرای هر وظیفه نقش تعیین‌کننده‌ای در کارایی کلی سیستم ایفا می‌کند.

در این مدل، مجموعه وظایف تولیدشده توسط دستگاه‌های اینترنت اشیا با نماد

$$T = \{t_1, t_2, \dots, t_n\}$$

نمایش داده می‌شود. این وظایف ممکن است از نظر حجم داده، میزان نیاز پردازشی و حساسیت به تأخیر با یکدیگر تفاوت داشته باشند. از سوی دیگر، مجموعه گره‌های لبه که مسئولیت اجرای این وظایف را بر عهده دارند با نماد

$$E = \{e_1, e_2, \dots, e_m\}$$

مشخص می‌شوند. هر گره لبه به‌صورت مستقل دارای ویژگی‌هایی نظیر توان پردازشی، ظرفیت انرژی و میزان بار پردازشی جاری است.

1. IoT devices
2. edge nodes
3. cloud servers

در این مدل، هر وظیفه می‌تواند روی هر گره لبه اجرا شود، اما دو هزینه‌ی اصلی دارد: انرژی مصرفی E_{ij} و تأخیر اجرا D_{ij}. هدف مدل، یافتن نگاشتی بهینه از وظایف به گره‌ها است؛ به‌گونه‌ای که مجموع انرژی مصرفی کل سیستم حداقل شود و در عین حال، هیچ وظیفه‌ای از آستانه مجاز تأخیر عبور نکند. برای این کار، یک متغیر دودویی x_{ij} تعریف شده است که مشخص می‌کند آیا وظیفه t_j به گره e_i اختصاص یافته است یا خیر.

وظایف بلادرنگ و حساس به تأخیر توسط دستگاه‌های IoT به گره‌های لبه (نزدیک به منبع داده) ارسال می‌شوند که باعث کاهش تأخیر و افزایش سرعت اجرا می‌گردند. در مقابل، وظایف سنگین و نیازمند پردازش پیچیده یا ذخیره‌سازی بلندمدت به سرورهای ابری منتقل می‌شوند (Karimi et al., 2022).

یکی از اجزای مهم در این معماری، سیستم زمان‌بندی وظایف[1] است که نقش کلیدی در کاهش تأخیر و بهبود کیفیت خدمات (QoS) ایفا می‌کند. این سیستم، به‌طور مداوم، اطلاعات مربوط به وضعیت صف‌های پردازشی، ظرفیت لحظه‌ای گره‌ها و اولویت وظایف را از گره‌های لبه دریافت کرده و بر اساس سیاست‌های بهینه‌سازی، تصمیم می‌گیرد که هر وظیفه در چه زمانی و بر روی کدام گره اجرا شود. در این فرایند، از الگوریتم‌های فراابتکاری هوشمند برای جست‌وجوی فضای راه‌حل و تعیین تخصیص مناسب وظایف استفاده می‌شود تا ضمن کاهش مجموع تأخیر اجرای وظایف، بار کاری نیز به‌صورت متعادل میان گره‌ها توزیع گردد (Kreutz et al., 2014).

ارتباط میان اجزای مختلف این مدل از طریق تبادل مداوم داده‌ها و تعامل مستمر میان گره‌ها و سیستم زمان‌بندی برقرار می‌شود. دستگاه‌های اینترنت اشیا داده‌های اولیه را به گره‌های لبه ارسال می‌کنند و این گره‌ها بر اساس وضعیت منابع و اولویت وظایف، آن‌ها را پردازش کرده یا در صورت نیاز به سطوح بالاتر ارسال می‌کنند. در همین حال، سیستم زمان‌بندی به‌طور پیوسته وضعیت منابع را پایش کرده و تخصیص وظایف را به‌صورت پویا به‌روزرسانی می‌کند. این سازوکار هماهنگ و پویا می‌تواند موجب کاهش تأخیر شبکه، افزایش بهره‌وری منابع محاسباتی و بهبود تجربه‌ی کاربران در کاربردهای بلادرنگی نظیر مراقبت‌های سلامت هوشمند، کنترل صنعتی و شهرهای هوشمند شود. در نتیجه، این مدل می‌تواند به‌عنوان چارچوبی مناسب برای مدیریت و زمان‌بندی وظایف در نسل آینده‌ی سامانه‌های هوشمند مبتنی بر اینترنت اشیا و پردازش لبه مورد استفاده قرار گیرد (Kazemi & Saberi, 2022; صادق و محمدمهدی، ۲۰۲۳).

شکل ۱. کلیت مدل سیستمی

1. task scheduling system

روش پیشنهادی

در این پژوهش، به‌منظور بهبود فرایند زمان‌بندی وظایف در محیط‌های پردازش لبه، یک چارچوب هوشمند مبتنی بر الگوریتم‌های فراابتکاری ارائه شده است. در این چارچوب، مسئله‌ی زمان‌بندی وظایف به‌عنوان یک مسئله‌ی بهینه‌سازی ترکیبی و چندهدفه مدل‌سازی می‌شود؛ به‌گونه‌ای که در آن باید مشخص شود که هر وظیفه در چه زمانی و بر روی کدام گره لبه اجرا گردد، به‌طوری‌که مجموع تأخیر اجرای وظایف در کل سیستم به حداقل رسیده، استفاده از منابع پردازشی بهینه شده و توزیع بار محاسباتی میان گره‌ها به‌صورت متعادل انجام شود.

در گام نخست، هر راه‌حل ممکن برای زمان‌بندی وظایف به‌صورت یک ساختار عددی کدگذاری می‌شود. در الگوریتم ژنتیک،[1] این ساختار به شکل کروموزوم نمایش داده می‌شود که هر ژن آن نشان‌دهنده‌ی گره مقصد برای اجرای یک وظیفه است. در الگوریتم بهینه‌سازی ازدحام ذرات[2] نیز هر راه‌حل به‌صورت موقعیت یک ذره در فضای چندبعدی تعریف می‌شود که در آن هر بعد، نمایانگر انتخاب یک گره پردازشی برای اجرای یکی از وظایف است.

به‌منظور ارزیابی کیفیت راه‌حل‌های تولید شده، یک تابع هدف[3] تعریف می‌شود که میزان کارایی هر زمان‌بندی را مشخص می‌کند. در این پژوهش، تابع هدف بر اساس حداقل‌سازی مجموع تأخیر اجرای وظایف در سیستم تعریف شده است. مقدار این تابع برای هر راه‌حل به‌صورت رابطه ۱ محاسبه می‌شود:

$$\boldsymbol{Fitness} = \min \sum_{i=1}^{n} \sum_{j=1}^{m} \boldsymbol{x_{ij}} . \boldsymbol{D_{ij}} \qquad (۱)$$

در این رابطه، x_{ij} یک متغیر دودویی است که نشان می‌دهد آیا وظیفه t_j به گره e_i تخصیص داده شده است یا خیر، و D_{ij} نیز بیانگر زمان لازم برای اجرای وظیفه t_j بر روی گره e_i است. هدف، یافتن ترکیبی از تخصیص‌ها است که مقدار این تابع را به حداقل برساند.

در فرایند جست‌وجوی راه‌حل‌های بهتر، الگوریتم‌های فراابتکاری از مجموعه‌ای از عملگرهای اکتشافی استفاده می‌کنند که امکان کاوش مؤثر فضای پاسخ را فراهم می‌آورند. در الگوریتم ژنتیک، این فرایند با الهام از اصول تکامل طبیعی انجام می‌شود. راه‌حل‌ها به‌صورت جمعیتی از کروموزوم‌ها نمایش داده می‌شوند و در هر نسل، دو عملگر اصلی برای تولید نسل جدید به‌کار گرفته می‌شود: نخستین عملگر، تقاطع[4] است که با ترکیب اطلاعات دو کروموزوم والد، فرزندانی جدید ایجاد می‌کند که ممکن است ویژگی‌های بهتری نسبت به والدین داشته باشند. عملگر دوم، جهش[5] است که با ایجاد تغییرات تصادفی کوچک در برخی ژن‌ها، تنوع جمعیت را افزایش می‌دهد. این عملگر نقش مهمی در جلوگیری از همگرایی زودرس الگوریتم به بهینه‌های محلی داشته و امکان کشف نواحی جدید در فضای جست‌وجو را فراهم می‌کند. تعامل مناسب این دو عملگر باعث می‌شود که الگوریتم ژنتیک به‌صورت تدریجی به سمت راه‌حل‌های با کیفیت بالاتر حرکت کند.

در حل مسائل بهینه‌سازی عملی، در نظر گرفتن محدودیت‌ها از اهمیت زیادی برخوردار است؛ زیرا بسیاری از راه‌حل‌های تولیدشده ممکن است از نظر ریاضی مناسب باشند، اما در عمل قابل پیاده‌سازی نباشند. در مدل پیشنهادی این پژوهش، دو محدودیت اصلی در فرایند زمان‌بندی در نظر گرفته شده است: نخست، محدودیت مهلت زمانی که بر اساس آن هر وظیفه باید در بازه‌ی زمانی مشخصی اجرا شود. دوم، محدودیت ظرفیت پردازشی گره‌ها که بیان می‌کند هر گره لبه تنها قادر است مقدار معینی از بار کاری را در هر لحظه پردازش کند. راه‌حل‌هایی که این

1. GA
2. PSO
3. fitness function
4. crossover
5. mutation

محدودیت‌ها را نقض کنند، یا از فرایند جست‌وجو حذف می‌شوند و یا با استفاده از تابع جریمه[1] مقدار شایستگی آن‌ها کاهش داده می‌شود تا احتمال انتخاب آن‌ها در نسل‌های بعدی کاهش یابد. این رویکرد موجب می‌شود الگوریتم تمرکز خود را بر روی راه‌حل‌هایی قرار دهد که علاوه بر کارایی بالا، از نظر عملی نیز قابل اجرا بوده و الزامات کیفیت خدمات را رعایت می‌کنند.

فرایند جست‌وجو در الگوریتم‌های فراابتکاری به‌صورت تکرارشونده انجام می‌شود؛ بدین معنا که در هر تکرار مجموعه‌ای از راه‌حل‌های جدید تولید شده، ارزیابی می‌شوند و بهترین آن‌ها به نسل بعدی منتقل می‌گردند. این روند تا زمانی ادامه می‌یابد که یکی از معیارهای توقف برقرار شود. این معیار می‌تواند شامل رسیدن به تعداد مشخصی از تکرارها، عدم بهبود قابل‌توجه در مقدار تابع هدف طی چند مرحله‌ی متوالی و یا دستیابی به یک مقدار قابل قبول برای تابع هدف باشد. چنین سازوکاری امکان کاوش مؤثر فضای جست‌وجو را فراهم کرده و در عین حال از مصرف بیش از حد زمان و منابع محاسباتی جلوگیری می‌کند.

در نهایت، عملکرد روش پیشنهادی از طریق اجرای سناریوهای شبیه‌سازی مورد ارزیابی قرار گرفته و معیارهایی مانند میانگین تأخیر اجرای وظایف مورد بررسی قرار گرفته‌اند. نتایج حاصل نشان می‌دهند که استفاده از الگوریتم‌های فراابتکاری در فرایند زمان‌بندی می‌تواند تخصیص وظایف را در گره‌های لبه به‌طور مؤثری مدیریت کرده و در عین حال با حفظ سطح مناسب کیفیت خدمات، منجر به کاهش قابل‌توجه تأخیر در سیستم شود (Saadi et al., 2020; حمیدرضا و محمدرضا، ۲۰۱۶).

ارزیابی و شبیه‌سازی

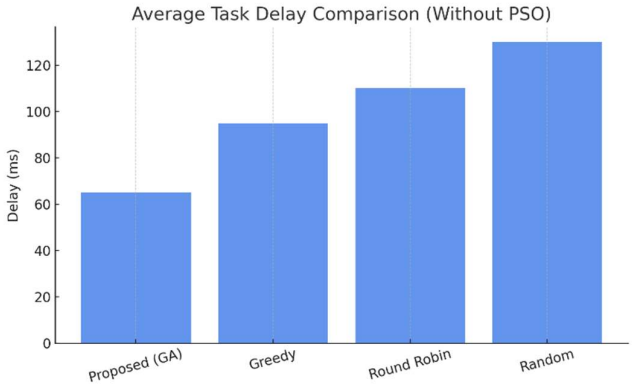
در این بخش، کارایی روش پیشنهادی مبتنی بر الگوریتم فراابتکاری ژنتیک (GA) از طریق شبیه‌سازی و در مقایسه با چند روش مرجع متداول شامل گریدی،[2] روند رابین[3] و رندوم[4] مورد ارزیابی قرار گرفته است. هدف از این مقایسه، بررسی میزان اثربخشی رویکرد پیشنهادی در بهبود شاخص‌های کلیدی عملکرد در محیط‌های پردازش لبه است. معیارهای اصلی مورد بررسی در این ارزیابی شامل میانگین تأخیر اجرای وظایف[5] و بهره‌وری منابع[6] در گره‌های لبه می‌باشند.

نمودار شکل ۲ میانگین تأخیر اجرای وظایف را برای چهار الگوریتم مورد مطالعه نمایش می‌دهد. شاخص میانگین تأخیر یکی از مهم‌ترین معیارها در سنجش کیفیت خدمات در سامانه‌های پردازش لبه به‌شمار می‌رود؛ زیرا بیانگر مدت‌زمانی است که یک وظیفه به‌طور متوسط برای انتظار در صف، تخصیص منابع و اجرای کامل صرف می‌کند. همان‌گونه که در این نمودار مشاهده می‌شود، الگوریتم پیشنهادی مبتنی بر ژنتیک با میانگین تأخیر ۶۵ میلی‌ثانیه بهترین عملکرد را در میان روش‌های مورد مقایسه ارائه داده است. این نتیجه نشان می‌دهد که الگوریتم ژنتیک توانسته است فرایند زمان‌بندی وظایف را به‌صورت هوشمندانه مدیریت کند و با در نظر گرفتن هم‌زمان عواملی نظیر اولویت وظایف، وضعیت صف‌ها، ظرفیت پردازشی گره‌ها و محدودیت‌های زمانی، تخصیص مناسبی از وظایف به گره‌های لبه انجام دهد.

در مقابل، الگوریتم‌های مرجع عملکرد ضعیف‌تری از خود نشان داده‌اند. روش گریدی با میانگین تأخیر ۹۵ میلی‌ثانیه، به‌دلیل تخصیص وظایف به اولین گره در دسترس بدون توجه به وضعیت لحظه‌ای بار گره‌ها، موجب افزایش

1. penalty function
2. greedy
3. round robin
4. random
5. average task delay
6. resource utilization

زمان انتظار برخی وظایف می‌شود. الگوریتم روند رابین نیز با میانگین تأخیر ۱۱۰ میلی‌ثانیه، اگرچه سعی در توزیع منصفانه وظایف میان گره‌ها دارد، اما به‌علت فقدان سازوکار هوشمند برای تطبیق با شرایط واقعی منابع، قادر به کاهش مؤثر تأخیر سیستم نیست. در نهایت، الگوریتم رندوم با ثبت میانگین تأخیر ۱۳۰ میلی‌ثانیه، بدترین عملکرد را به خود اختصاص داده است که این امر ناشی از تخصیص کاملاً تصادفی وظایف بدون در نظر گرفتن هیچ‌گونه معیار بهینه‌سازی یا اطلاعات سیستمی است.

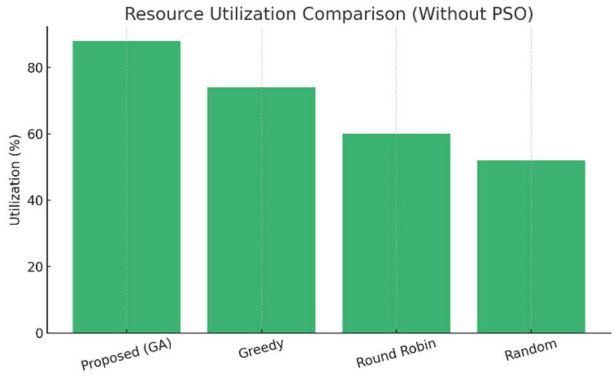


شکل ۲. تأخیر

نتایج حاصل از این ارزیابی به‌روشنی نشان می‌دهند که به‌کارگیری الگوریتم‌های هوشمند و مبتنی بر جست‌وجوی فراابتکاری، به‌ویژه الگوریتم ژنتیک، می‌تواند نقش بسزایی در کاهش تأخیر سیستم‌های توزیع‌شده و بلادرنگ ایفا کند و از منظر عملیاتی، کارایی به‌مراتب بالاتری نسبت به روش‌های سنتی و ساده ارائه دهد.

نمودار شکل ۳ میزان بهره‌وری منابع را برای همان چهار الگوریتم نشان می‌دهد. این شاخص بیانگر درصد استفاده مؤثر از ظرفیت پردازشی گره‌های لبه در طول زمان است. بهره‌وری بالاتر نشان‌دهنده‌ی آن است که منابع محاسباتی سیستم به‌صورت متعادل و کارآمد مورد استفاده قرار گرفته‌اند و از بیکارماندن یا استفاده‌ی ناکافی از گره‌ها جلوگیری شده است. بر اساس نتایج نمایش‌داده‌شده در این نمودار، الگوریتم پیشنهادی ژنتیک با ثبت بهره‌وری ۸۸٪، بالاترین میزان استفاده‌ی مؤثر از منابع را در میان الگوریتم‌های مورد بررسی داشته است. این موضوع نشان می‌دهد که زمان‌بندی انجام‌شده توسط الگوریتم ژنتیک نه‌تنها موجب کاهش تأخیر شده، بلکه استفاده‌ای متوازن و هدفمند از ظرفیت پردازشی گره‌های لبه را نیز فراهم کرده است.

در این میان، روش گریدی با بهره‌وری ۷۴٪ در رتبه‌ی دوم قرار می‌گیرد. اگرچه این روش در برخی موارد گره‌های مناسب‌تری را برای اجرای وظایف انتخاب می‌کند، اما به‌دلیل عدم توجه به توزیع بار و اولویت وظایف، قادر به دستیابی به بهره‌وری حداکثری نیست. الگوریتم روند رابین نیز با بهره‌وری ۶۰٪، به‌سبب تخصیص یکنواخت اما غیربهینه وظایف، منجر به استفاده‌ی پراکنده و کم‌اثر از منابع می‌شود. در نهایت، الگوریتم رندوم با بهره‌وری ۵۲٪ پایین‌ترین عملکرد را از خود نشان داده و به‌وضوح بیانگر آن است که تخصیص تصادفی وظایف، به‌ویژه در محیط‌های با منابع محدود و حساس به تأخیر، منجر به ناکارآمدی شدید در سیستم خواهد شد.



شکل ۳. مصرف منابع

جدول ۱. نتایج شبیه‌سازی: مقایسه تأخیر و بهره‌وری منابع الگوریتم‌ها (Kreutz et al., 2014; Soltani et al., 2016)

الگوریتم	میانگین تأخیر (میلی‌ثانیه)	بهره‌وری منابع (٪)	توضیح ترکیبی
ژنتیک (پیشنهادی)	۶۵ ms	٪۸۸	بهترین عملکرد در هر دو معیار؛ زمان‌بندی هوشمند با درنظرگرفتن اولویت وظایف، وضعیت صف‌ها و ظرفیت گره‌ها؛ منجر به کاهش محسوس تأخیر و استفاده متوازن و بهینه از ظرفیت گره‌های لبه می‌شود.
گریدی	۹۵ ms	٪۷۴	تخصیص به «اولین گره مناسب»؛ بهره‌وری نسبتاً خوب ولی بدون تعادل کامل بار. نبود نگاه سراسری به وضعیت بار باعث افزایش تأخیر در برخی گره‌ها و استفاده غیربهینه از تعدادی از منابع می‌شود.
روند رابین	۱۱۰ ms	٪۶۰	توزیع یکنواخت ولی غیرهوشمند؛ نادیده‌گرفتن وضعیت لحظه‌ای بار گره‌ها. اگرچه تخصیص ظاهراً منصفانه است، اما به دلیل عدم انطباق با وضعیت واقعی منابع، هم تأخیر بالاتر و هم بهره‌وری پایین‌تر نسبت به GA مشاهده می‌شود.
رندوم	۱۳۰ ms	٪۵۲	تخصیص کاملاً تصادفی، بدون هرگونه بهینه‌سازی. این رویکرد منجر به بیشترین میانگین تأخیر و پایین‌ترین بهره‌وری منابع می‌شود و ناکارآمدی روش‌های فاقد منطق زمان‌بندی را در محیط‌های IoT–Edge به‌خوبی نشان می‌دهد.

جدول ۲. ویژگی‌های الگوریتم‌های فرااِبتکاری استفاده‌شده (Soleymani & Nematzadeh, 2017)

الگوریتم	نمایش راه‌حل	مکانیزم جست‌وجو	عملگرهای کلیدی	نقش در زمان‌بندی
(ژنتیک)	کروموزوم؛ هر ژن نشان‌دهنده گره مقصد یک وظیفه	تکامل جمعیت راه‌حل‌ها بر اساس انتخاب طبیعی	تقاطع (Crossover)، جهش (Mutation)، انتخاب	یافتن ترکیب‌های خوب تخصیص وظایف به گره‌های لبه، جلوگیری از گیر افتادن در بهینه‌های محلی
(ازدحام ذرات)	ذره در فضای چندبعدی؛ هر بعد نشان‌دهنده‌ی گره تخصیص برای یک وظیفه	حرکت ذرات بر مبنای بهترین تجربه فردی و جمعی	به‌روزرسانی سرعت و موقعیت ذرات	جست‌وجوی نقطه‌ای در فضای تخصیص وظایف؛ همگرایی سریع به راه‌حل‌های با تأخیر پایین

معماری، لایه‌بندی و نقش پردازش لبه در زمان‌بندی وظایف IoT

در این پژوهش، مدل معماری بر پایه‌ی سه لایه‌ی اینترنت اشیا–لبه–ابری بنا شده است. لایه‌ی نخست شامل دستگاه‌های اینترنت اشیا به‌عنوان تولیدکنندگان داده‌های بلادرنگ، لایه‌ی دوم شامل گره‌های لبه به‌عنوان نزدیک‌ترین واحد پردازشی به منبع داده و لایه‌ی سوم شامل زیرساخت ابری برای پردازش‌های سنگین‌تر و ذخیره‌سازی بلندمدت است.

ویژگی متمایز این مقاله نسبت به سایر پژوهش‌های مشابه، افزودن یک سیستم زمان‌بندی پویا و هوشمند در لایه لبه است که نقش یک کنترل‌کننده مرکزی را ایفا می‌کند. این کنترل‌کننده به‌طور مداوم وضعیت منابع پردازشی، طول

صف‌ها و مهلت زمانی وظایف را ارزیابی کرده و تعیین می‌کند هر وظیفه در چه زمانی و بر روی کدام گره لبه اجرا شود تا ضمن کاهش تأخیر، کیفیت خدمات از طریق تخصیص بهینه منابع ارتقا یابد.

در مقابل، پژوهش‌های پیشین، به‌ویژه چارچوب‌های مبتنی بر شبکه‌های نرم‌افزارمحور (SDN)، معماری خود را عمدتاً بر پایه‌ی مدیریت جریان داده و کنترل ترافیک شبکه بنا کرده‌اند که ماهیتی شبکه‌محور دارند، در حالی‌که معماری مقاله‌ی حاضر محاسبه‌محور و وظیفه‌محور است و تمرکز آن بر مسئله‌ی زمان‌بندی کارآمد در لبه می‌باشد.

در ساختار پیشنهادی، دستگاه‌های IoT وظایف خود را با مشخصاتی نظیر حجم داده، مهلت زمانی و نیازمندی‌های QoS به گره‌های لبه ارسال می‌کنند و گره‌های لبه در مرحله زمان‌بندی باید سه مؤلفه محدودیت‌های پردازشی و حافظه‌ای هر گره، محدودیت‌های زمانی وظایف و سربار ارتباطی در صورت نیاز به ابر را همزمان مدنظر قرار دهند.

بررسی جدول ۳ نشان می‌دهد معماری‌های این حوزه به‌طور کلی در دو طبقه تک‌مرکزی (تصمیم‌گیری متمرکز در ابر یا کنترل‌کننده شبکه) و توزیع‌شده (گره‌های لبه دارای هوشمندی محلی) قرار می‌گیرند. معماری مقاله حاضر در گروه دوم جای می‌گیرد؛ به گونه‌ای که هر گره لبه همانند یک عامل هوشمند در تعامل با سایر گره‌ها و وضعیت جاری شبکه، بهترین تصمیم را برای تخصیص وظایف اتخاذ می‌کند. چنین ساختاری بستر مناسبی برای استفاده از الگوریتم‌های فراابتکاری جمعیت‌محور، مانند GA و PSO، فراهم می‌سازد؛ زیرا هر راه‌حل در این الگوریتم‌ها یک سناریوی تخصیص وظایف است و جمعیت راه‌حل‌ها مجموعه‌ای از حالات ممکن را در فضای جست‌وجو نمایندگی می‌کنند. به این ترتیب، لایه لبه در این معماری نه‌تنها محل اصلی اجرای وظایف است، بلکه نقش مرکز تصمیم‌گیری و مدیریت منابع را نیز بر عهده دارد (Balaji et al., 2022).

جدول ۳. مقایسه برخی کارهای پیشین از نظر معماری و لایه‌بندی و نوع یکپارچه‌سازی IoT–Blockchain–Edge–SDN (Alfawaz, 2022; Hashembeigi et al., 2023; Mirtaheri, 2023; Mohaghar et al., 2021; Tabatabaei, 2023)

نوع معماری	لایه‌های اصلی	نقش IoT	نقش بلاکچین[1]	نقش لبه/ مه	نوآوری کلیدی
SDN یکپارچه با IoT و NFV	کنترل، مجازی‌سازی، امنیت	تولید جریان و ترافیک	تشدید چالش‌های مقیاس‌پذیری	تصمیم‌گیری زمان‌بندی	چارچوب SDN برای بهینه‌سازی جریان‌ها
لبه + بلاکچین+IoT	کنترل، زمان‌بندی، اعتبارسنج	تولید تسک	اعتبارسنجی تراکنش	اجرای زمان‌بندی	الگوریتم نخبه‌گرا
لایه سه‌تایی لبه–مه–ابر	سرویس‌دهنده، مه، کاربر	تولید درخواست	ثبت تراکنش	محل زمان‌بندی	MINLP + TLBO بدون پارامتر
مشابه چیدمان لبه مدل‌های IoT	ماشین‌ها، قطعات، مسیر	داده سنسورها	ثبت رویدادها	کاهش تراکم	ITS با جستجوی ممنوعه پویا
حسگر، خوشه، چاهک	حسگر، خوشه، چاهک	ارسال داده	ثبت تصمیمات خوشه‌بندی	بهینه‌سازی انرژی	خوشه‌بندی KH با چاهک متحرک
Fog + IoT + Cloudlet	مه، کلودلت، کاربر	جریان کاری	ثبت تخصیص	تصمیم تخصیص	بهبود ۲۲٪ انرژی و ۴۷٪ زمان پاسخ
IoTکتابخانه + بلاکچین	ادراک، شبکه، مدیریت	حسگر محیط/رفیع	ثبت چرخه عمر	ذخیره‌سازی/تحلیل	IoTکتابخانه سبز[2] +
IoT– سلامت– بلاکچین	ادراک، شبکه، کاربرد	دریافت داده حیاتی	ثبت نسخه سلامت	تحلیل لبه	کاهش زمان پاسخ ۲۰–۳۰٪
IoT مسیریابی	گره، مسیر، sink	ترافیک حسگر	ثبت انرژی	مسیریابی کم‌هزینه	FLGOA برای افزایش عمر شبکه

1. blockchain
2. green library

نقش الگوریتم‌های فراابتکاری GA، PSO و دیگر روش‌ها در بهینه‌سازی زمان‌بندی در لبه

جدول ۲ به مقایسه‌ی الگوریتم‌های فراابتکاری مطرح شده، نظیر GA، PSO، GWO و ABC، از منظر نوع جست‌وجو، سازوکار عملکرد و اهداف بهینه‌سازی پرداخته است. از آنجا که مسئله‌ی زمان‌بندی وظایف در این پژوهش در دسته‌ی مسائل بهینه‌سازی ترکیبی و چندهدفه قرار دارد و فضای جست‌وجوی آن به دلیل تعداد زیاد وظایف و گره‌ها بسیار گسترده و پیچیده است، روش‌های قطعی قادر به بررسی کامل تمام حالت‌های ممکن نخواهند بود. به همین دلیل، استفاده از الگوریتم‌های فراابتکاری جمعیت‌محور که با تولید و تکامل مجموعه‌ای از راه‌حل‌ها از گرفتار شدن در بهینه‌های محلی جلوگیری می‌کنند، رویکردی مناسب برای جست‌وجوی کارآمد در این فضا محسوب می‌شوند.

در الگوریتم ژنتیک، هر راه‌حل به‌صورت یک کروموزوم نمایش داده می‌شود و عملگرهای تقاطع و جهش برای تولید نسل‌های جدید به کار می‌روند. مزیت اصلی ژنتیک توانایی آن در تولید ترکیب‌های متنوع از تخصیص وظایف به منابع است؛ به‌گونه‌ای که می‌تواند ویژگی‌های مطلوب راه‌حل‌های مختلف را با یکدیگر ترکیب کند. در مقابل، الگوریتم ازدحام ذرات بر مبنای حرکت ذرات در فضای جست‌وجو عمل می‌کند؛ جایی که هر ذره با استفاده از بهترین تجربه‌ی فردی[1] و بهترین تجربه‌ی جمعی[2] موقعیت خود را به‌روزرسانی می‌کند. این سازوکار به ازدحام ذرات امکان می‌دهد با سرعت نسبتاً بالایی نواحی مناسب فضای جست‌وجو را شناسایی کند، هرچند در صورت تنظیم نامناسب پارامترها احتمال همگرایی زودرس وجود دارد.

انتخاب GA و PSO در روش پیشنهادی این پژوهش دلایل متعددی دارد: نخست، ماهیت گسسته و بسیار بزرگ فضای جست‌وجو در مسئله‌ی زمان‌بندی وظایف در محیط لبه است که GA با توانایی خود در تولید ترکیب‌های متنوع پاسخ‌گوی مناسبی برای آن محسوب می‌شود. دوم، نیاز به تبادل اطلاعات در سطح کل سیستم است که PSO با بهره‌گیری از مفهوم بهترین تجربه‌ی جمعی دید سراسری نسبت به وضعیت فضای جست‌وجو فراهم می‌کند. سوم، ماهیت چندهدفه مسئله است؛ هرچند تابع هدف اصلی کمینه‌سازی مجموع تأخیر تعریف شده، اما وجود قیود عملی مانند ظرفیت گره‌ها و مهلت زمانی وظایف باعث می‌شود که مسئله در عمل اهداف ضمنی دیگری، مانند افزایش سطح رضایت QoS و جلوگیری از سربار منابع، را نیز شامل شود.

در خصوص مدیریت قیود، مقاله اشاره می‌کند که محدودیت‌های مربوط به مهلت زمانی وظایف و ظرفیت منابع می‌توانند به دو روشِ حذف راه‌حل‌های نامعتبر یا اعمال تابع جریمه در تابع هدف مدیریت شوند. رویکرد تابع جریمه که در الگوریتم‌هایی مانند GA و PSO رایج است، به الگوریتم اجازه می‌دهد در مراحل اولیه‌ی جست‌وجو از راه‌حل‌هایی که اندکی قیود را نقض کرده‌اند نیز برای کشف نواحی مناسب استفاده کند و سپس به‌تدریج جمعیت را به سمت پاسخ‌های مقید و بهینه هدایت نماید. شایان ذکر است که هر دو الگوریتم GA و PSO از نوع جمعیت‌محور هستند؛ به این معنا که به جای تمرکز بر یک راه‌حل منفرد، مجموعه‌ای از راه‌حل‌ها را هم‌زمان تکامل می‌بخشند. این ویژگی برای مسئله زمان‌بندی در محیط‌های پویای IoT و لبه که شرایط سیستم نظیر میزان بار گره‌ها و نرخ ورود وظایف مرتباً در حال تغییر است، اهمیت بسزایی دارد؛ زیرا وجود چندین راه‌حل در جمعیت الگوریتم بیانگر چند سناریوی مختلف زمان‌بندی است که می‌توانند خود را با شرایط متغیر سیستم سازگار کنند. در نهایت، GA به دلیل تنوع جمعیت و عملگر جهش، توانایی بالایی در اکتشاف فضای جست‌وجو دارد، در حالی‌که PSO با بهره‌گیری از

1. pbest
2. gbest

بهترین تجربه‌های فردی و جمعی در بهره‌برداری از نواحی مناسب کشف‌شده، عملکرد مؤثری از خود نشان می‌دهد، به همین دلیل ترکیب این دو الگوریتم به‌عنوان رویکردی هیبریدی در بسیاری از پژوهش‌ها مورد استفاده قرار می‌گیرد.

جدول ۴. مقایسه پژوهش‌هایی از نظر الگوریتم فراابتکاری، نوع جست‌وجو و ویژگی‌های بهینه‌سازی(Khadem et al., 2023)

نوع الگوریتم	نوع جست‌وجو	ویژگی‌های الگوریتم	اهداف بهینه‌سازی	مزیت کلیدی
PSO /GA (پیشنهادی)	جمعیت‌محور	سازگار با SDN	تأخیر، صف، انرژی	کنترل جریان چندمسیره
نخبه‌گرا (Elite)	انتخاب نخبه + تنوع	۵–۱۵٪ جمعیت نخبه	تأخیر، انرژی، سربار	بهبود ۱۰–۳۰٪
Teacher–Learner	بدون پارامتر	۲ فاز آموزش/یادگیری	سود، QoS	بهبود سود ۱۲۵→۳۰۰
Tabu + Adaptive	همسایگی افزایشی	ممنوعه پویا	min_sum ،min_max	تراکم کمتر
KH + چاهک متحرک	حرکت جمعی	جست‌وجوی سریع	انرژی، تأخیر	بهبود ۴۶٪ کل
ALO	شکار–محاصره	فازی + چندمعیاره	انرژی، تأخیر، بار	۲۲٪ انرژی کمتر
ملخ فازی[1]	فازی + ازدحامی	جست‌وجوی تطبیقی	انرژی باقیمانده	بهبود ۹–۱۶٪
ازدحامی	تنظیم خودکار	کمینه‌سازی تأخیر	مسیر بهینه	کاهش هزینه ۱۰–۲۰٪
چندهدفه	Pareto	زمان‌بندی چندهدفه	زمان، هزینه، بار	کاهش makespan ۲۰–۳۰ ٪

نتایج کمی، درصد بهبود و اهمیت طراحی سناریوهای آزمایشی در زمان‌بندی لبه

جدول ۵ نسبت به دو جدول پیشین، تمرکز خود را از سطح معماری و ویژگی‌های الگوریتمی به ارزیابی کمی و نتایج عددی معطوف کرده است. این جدول نشان می‌دهد که هر پژوهش در چه سناریوی آزمایشی ارزیابی شده، از چه معیارهایی برای سنجش عملکرد استفاده کرده و چه میزان بهبود نسبت به روش‌های مرجع گزارش شده است. به این ترتیب، جدول ۵ حلقه‌ی اتصال میان مدل‌سازی نظری و کارایی عملی محسوب می‌شود.

در پژوهش حاضر، تابع هدف اصلی کمینه‌سازی مجموع تأخیر اجرای وظایف است و قیود اساسی شامل مهلت زمانی وظایف و ظرفیت پردازشی گره‌های لبه می‌باشند. از این‌رو، معیارهای اندازه‌گیری بر شاخص‌هایی نظیر تأخیر انتها به انتها، زمان پاسخ و معیارهای QoS متمرکز هستند. هرچند در برخی از پژوهش‌های مشابه معیارهایی مانند مصرف پهنای باند نیز گزارش می‌شوند، اما در مقاله‌ی حاضر تمرکز اصلی بر تأخیر، کیفیت خدمات و بهره‌وری منابع است.

یکی از مهم‌ترین بخش‌های تحلیلی جدول ۵، ستون سناریوی آزمایش است؛ زیرا اعداد خام یا درصدهای بهبود زمانی معنا می‌یابند که بدانیم نتایج در چه شرایطی حاصل شده‌اند. تحلیل سناریوی آزمایش باید سه بُعد کلیدی را روشن کند: نخست، نوع بار کاری که الگوی ورود وظایف تأثیر مستقیمی بر عملکرد الگوریتم دارد؛ دوم، تعداد گره‌های لبه که با افزایش آن، فضای جست‌وجوی مسئله گسترش یافته و مقیاس‌پذیری روش پیشنهادی آشکار می‌شود و سوم، نرخ موفقیت در رعایت مهلت زمانی که بیانگر توان الگوریتم در حفظ QoS در شرایط بار بالا است.

در ستون نتایج کمی، مقادیر خام معیارهای اندازه‌گیری ارائه می‌شود و در ستون درصد بهبود، امکان مقایسه‌ی نسبی میان مطالعات مختلف فراهم می‌گردد؛ زیرا درصد بهبود مستقل از مقیاس مطلق، تصویری روشن‌تر از کیفیت روش ارائه می‌دهد. از منظر بهینه‌سازی، اگر الگوریتم‌های فراابتکاری تنها بهبودی اندک ایجاد کنند، هزینه‌ی محاسباتی آن‌ها توجیه‌پذیر نخواهد بود، اما بهبود ۲۰ تا ۳۰ درصدی می‌تواند هزینه‌ی اضافی را منطقی جلوه دهد. همچنین، پارامترهای الگوریتمی نظیر اندازه‌ی جمعیت، تعداد تکرارها و معیار توقف تأثیر مستقیمی بر زمان اجرا و کیفیت پاسخ

1. fuzzy grasshopper

نهایی دارند و در کاربردپذیری روش در محیط‌های لبه که فرایند زمان‌بندی باید سریع و کم‌هزینه باشد، نقشی کلیدی ایفا می‌کنند.

در مجموع، جدول ۵ پیوند نهایی میان نظریه‌ی زمان‌بندی فراابتکاری و عملکرد واقعی در محیط‌های لبه‌محور را برقرار می‌سازد و نشان می‌دهد که روش پیشنهادی تا چه حد در کاربردهای عملی قابلیت اتکا دارد (Yeganeh & Vaezpour, 2018).

جدول ۵. مقایسه مقاله‌ها از نظر نتایج کمی، درصد بهبود و پارامترهای آزمایشی (Mahmoudi et al., 2021)

سناریو آزمایش	معیارهای اندازه‌گیری	نتایج کمی	درصد بهبود
SDN-IoT	تأخیر، کنترل	—	—
Benchmark Functions	هدف چندمعیاره	بهبود ۱۰–۳۰٪	۳۰٪ بهتر
Cloud–Fog	سود، تأخیر، رضایت	سود ۱۲۵→۳۰۰	+۴۰٪
۵۰ ماشین	time ،min_sum	از ۱۰۳→۹۵	۸–۱۰٪
IoT Routing	انرژی، تأخیر	انرژی −۱۲٫۷٪	+14% throughput
IoT–Fog	انرژی، تأخیر، زمان	زمان پاسخ −۴۷٪	انرژی −۲۲٪
IoT Routing	مرگ ۳۰٪ نود	+۹ تا ۱۶٪ بهتر	انرژی +۲۱٪
Health IoT	دقت، زمان پاسخ	دقت >۹۰٪	زمان پاسخ −۳۰٪
IoT–Cloud	هزینه ،makespan	−۲۰–۳۰٪ زمان کار	منابع +۲۵٪

قابلیت تعمیم، حوزه‌های کاربرد و جایگاه معماری Edge–IoT مقاله در میان سامانه‌های هوشمند

جدول ۶ به پرسش بنیادین قابلیت تعمیم معماری و الگوریتم زمان‌بندی پیشنهادی به سایر حوزه‌های کاربردی مبتنی بر IoT و پردازش لبه می‌پردازد. حوزه‌هایی همچون مراقبت سلامت، کنترل صنعتی، شهر هوشمند و حمل‌ونقل هوشمند، همگی ویژگی مشترک تولید داده‌های بلادرنگ و حساسیت شدید به تأخیر را دارند؛ بنابراین، منطق اصلی مسئله، یعنی زمان‌بندی کارآمد وظایف در لبه، برای کاهش تأخیر و بهبود QoS در میان این حوزه‌ها مشترک است.

ساختار مسئله در پژوهش حاضر به صورت عمومی طراحی شده و به سناریوی خاصی وابسته نیست. در مدل زمان‌بندی، هر وظیفه با پارامترهایی نظیر حجم داده، مهلت زمانی و سطح QoS مشخص می‌شود و این مدل هیچ‌گونه وابستگی به نوع خاصی از داده یا کاربرد ندارد. از این‌رو، فرمول‌بندی ریاضی مسئله در حوزه‌های مختلف ثابت می‌ماند و تنها سناریوهای ورود وظایف، رفتار شبکه یا الگوی بار کاری تغییر می‌کنند.

الگوریتم‌های GA و PSO نیز ذاتاً مستقل از حوزه‌ی کاربردی هستند. این روش‌های فراابتکاری تا زمانی‌که مسئله به تابع هدف، قیود و نمایش راه‌حل قابل تبدیل باشد، در حوزه‌های حمل‌ونقل، سلامت، انرژی، شهر هوشمند و ده‌ها محیط دیگر قابل به‌کارگیری هستند. ستون «استفاده از فراابتکاری‌ها» در جدول ۶ نیز موفقیت این گروه الگوریتم‌ها را در دامنه‌ی گسترده‌ای از کاربردها تأیید می‌کند.

معماری سه‌لایه IoT–Edge–Cloud نیز ساختاری است که در اغلب کاربردهای مبتنی بر داده‌های بلادرنگ استفاده می‌شود؛ از مراقبت سلامت تا کنترل صنعتی و شبکه‌های حمل‌ونقل، این الگو ستون فقرات سیستم را تشکیل می‌دهد. لایه لبه در همه کاربردها مسئول کاهش تأخیر، افزایش کیفیت خدمات و اجرای محلی وظایف است و تفاوت‌ها تنها در نوع داده، سیاست‌های اولویت‌دهی و شدت بار کاری دیده می‌شود، نه در نقش بنیادی لایه‌ها.

از سوی دیگر، اگرچه در مقاله نقش بلاکچین یا ماژول‌های امنیتی به‌صورت محدود مطرح شده است، اما جدول ۶ نشان می‌دهد که اکثر معماری‌های Edge–IoT قابلیت ترکیب با سازوکارهای امنیتی و حسابرسی را دارند.

زمان‌بندی وظایف یک جزء مستقل و ماژولار است و حتی با افزودن سیستم‌های امنیت، ثبت تراکنش یا مدیریت اعتماد، ساختار اصلی زمان‌بندی تغییری نمی‌کند که این استقلال ماژولار، قابلیت تعمیم معماری را افزایش می‌دهد.

در نهایت، جدول ۶ به تمایز مهم میان سطح مفهومی قابل تعمیم معماری IoT–Edge–Cloud به‌همراه مدل زمان‌بندی و الگوریتم فراابتکاری (و سطح اجرایی خاص هر حوزه) پارامترهای بار، نوع داده‌ها، توزیع arrival، پیکربندی شبکه و قیود عملیاتی اشاره می‌کند. در مقاله‌ی حاضر، عمده‌ی طراحی در سطح مفهومی انجام شده است. بنابراین، برای به‌کارگیری این چارچوب در حوزه‌ای دیگر، نیازی به بازنگری اساسی مدل نیست و تنها تنظیم پارامترهای سناریو بر اساس کاربرد جدید کافی است. این ویژگی یکی از معیارهای اصلی «قابلیت تعمیم بالا» محسوب می‌شود.

جدول ۶. مقایسه مقاله‌ها از منظر قابلیت تعمیم به معماری‌های Blockchain–IoT–Edge و حوزه کاربرد (Khadem et al., 2023; Kreutz et al., 2014; Tabatabaei, 2023; منفرد و همکاران، ۲۰۲۳)

حوزه کاربرد	نقش Blockchain	نقش Edge/Fog	استفاده پیشنهادی از Metaheuristics	نتیجه تعمیم
SDN–IoT	ثبت مسیر	زمان‌بندی	PSO/GA	کاهش سربار کنترل
Edge–Blockchain	انتخاب ولیدیتور	نگاشت تسک	نخبه‌گرا	توزیع بار
Fog	ثبت تراکنش	تخصیص منابع	TLBO	بهینه‌سازی سود/QoS
Layout–IoT	ضبط مسیرها	کنترل تراکم	ITS	پایداری تراکم
Sensor Network	ثبت خوشه	مدیریت انرژی	KH	اعتماد + انرژی
Fog Computing	ثبت سیاست	migration	ALO	SLA + انرژی
IoT Routing	ثبت انرژی	انتخاب مسیر	FLGOA	افزایش عمر شبکه
Smart Library	ثبت چرخه عمر	تحلیل لبه	PSO/DE	Green Library
Health	consent ledger	محاسبه لبه	PSO/GA	کاهش هزینه
IoT Security	کاهش spoofing	resilient routing	بهینه‌سازی اجماع	امنیت ۳۰–۴۰٪ بهتر
Banking	ثبت احراز هویت	کنترل ریسک	evolutionary	کاهش ریسک
Cloud–IoT	ردپای تسک	کاهش بار	NSGA-II/PSO	بهبود ۳۰٪

نتیجه‌گیری

رشد روزافزون اینترنت اشیا، سامانه‌های هوشمند و کاربردهای بلادرنگ موجب شده است که معماری‌های سنتی مبتنی بر رایانش ابری به‌تنهایی پاسخ‌گوی نیازهای پردازشی جدید نباشند. در چنین شرایطی، پردازش لبه به‌عنوان یک راهکار کارآمد برای کاهش تأخیر، کاهش وابستگی به ابر و بهبود کیفیت خدمات مطرح شده است. با این حال، محدودیت منابع پردازشی و انرژی در گره‌های لبه، مسئله زمان‌بندی وظایف را به یکی از مهم‌ترین چالش‌های این حوزه تبدیل می‌کند. هدف اصلی این پژوهش، بررسی کارآمدی الگوریتم‌های فراابتکاری در بهینه‌سازی فرایند زمان‌بندی وظایف در محیط‌های IoT–Edge و تحلیل نقش آن‌ها در کاهش تأخیر و افزایش بهره‌وری منابع بوده است.

در این مقاله، یک معماری سه‌لایه شامل دستگاه‌های اینترنت اشیا، گره‌های لبه و زیرساخت ابری ارائه شد که در آن، سیستم زمان‌بندی هوشمند وظیفه‌ی تخصیص بهینه وظایف به گره‌های پردازشی را بر عهده دارد. مسئله‌ی زمان‌بندی به‌صورت یک مسئله بهینه‌سازی ترکیبی و چندهدفه مدل‌سازی شد و الگوریتم‌های فراابتکاری، به‌ویژه الگوریتم ژنتیک (GA) و بهینه‌سازی ازدحام ذرات (PSO)، برای جست‌وجوی فضای پاسخ مورد استفاده قرار گرفتند. این الگوریتم‌ها با قابلیت جست‌وجوی گسترده در فضای راه‌حل و جلوگیری از گرفتار شدن در بهینه‌های محلی، توانستند تخصیص مناسب‌تری از وظایف را نسبت به روش‌های سنتی ارائه دهند.

نتایج شبیه‌سازی نشان داد که روش پیشنهادی مبتنی بر الگوریتم ژنتیک، در مقایسه با الگوریتم‌های Greedy، Round Robin و Random، عملکرد بهتری در هر دو معیار میانگین تأخیر و بهره‌وری منابع دارد. کاهش میانگین تأخیر به ۶۵ میلی‌ثانیه و دستیابی به بهره‌وری ۸۸ درصدی منابع نشان می‌دهد که استفاده از الگوریتم‌های فراابتکاری می‌تواند نقش مهمی در بهبود عملکرد سامانه‌های پردازش لبه ایفا کند. همچنین، نتایج بیانگر آن بودند که رویکردهای هوشمند مبتنی بر جست‌وجوی جمعیت‌محور، توانایی بیشتری در توزیع متعادل بار کاری میان گره‌ها و حفظ کیفیت خدمات در شرایط پویا دارند.

از دیگر یافته‌های مهم این پژوهش، قابلیت تعمیم معماری پیشنهادی به حوزه‌هایی نظیر سلامت هوشمند، شهرهای هوشمند، حمل‌ونقل هوشمند و سامانه‌های صنعتی است. ساختار ماژولار مدل ارائه‌شده این امکان را فراهم می‌کند که در آینده، فناوری‌هایی مانند بلاکچین، مکانیزم‌های امنیتی و روش‌های یادگیری ماشین نیز به آن افزوده شوند. در مجموع، نتایج این پژوهش نشان می‌دهند که ترکیب پردازش لبه با الگوریتم‌های فراابتکاری، رویکردی مؤثر برای مدیریت هوشمند منابع و زمان‌بندی وظایف در سامانه‌های نسل آینده اینترنت اشیا محسوب می‌شود و می‌تواند زمینه‌ساز توسعه زیرساخت‌های مقیاس‌پذیر، کم‌تأخیر و قابل اعتماد در محیط‌های هوشمند باشد.

منابع

حمیدرضا، ی. پ.، و محمدرضا، ح. آ. (۲۰۱۶). اینترنت اشیا (IoT): کاربردها، فناوری‌ها و چالش‌های مورد بحث.

صادق، ع. آ.، . محمدمهدی، م. (۲۰۲۳). اولویت‌بندی حوزه‌های کاربردی اینترنت اشیا در صنایع ایران.

منفرد، ط.، نورالدین، شایان، و قطری، ر. ز. (۲۰۲۳). بررسی اثر پیاده‌سازی اینترنت اشیا بر صنعت حمل ریلی بار با رویکرد آینده‌پژوهی. *فصلنامه فناوری اطلاعات و ارتباطات ایران*، ۵۵(۵۵)، ۱۹۱.

Abdelmoneem, R. M., Benslimane, A., & Shaaban, E. (2020). Mobility-aware task scheduling in cloud-Fog IoT-based healthcare architectures. *Computer Networks*, *179*, 107348.

Alfawaz, K. M. (2022). IOT Future Security Challenges and Recent Solutions. *Journal of Information Technology Management*, *14*(2), 1-14.

Aly, M., Khomh, F., & Yacout, S. (2021). What do practitioners discuss about iot and industry 4.0 related technologies? characterization and identification of iot and industry 4.0 categories in stack overflow discussions. *Internet of Things*, *14*, 100364.

Balaji, S. S., Ragunathan, S., Visagavel, K., & Ganesh Kumar, S. (2022). Design and optimization of layout problem using meta-heuristic algorithms. *Scientia Iranica*, *29*(5), 2290-2303.

Bittencourt, L. F., Diaz-Montes, J., Buyya, R., Rana, O. F., & Parashar, M. (2017). Mobility-aware application scheduling in fog computing. *IEEE Cloud Computing*, *4*(2), 26-35.

Dastjerdi, A. V., Gupta, H., Calheiros, R. N., Ghosh, S. K., & Buyya, R. (2016). Fog computing: Principles, architectures, and applications. In *Internet of Things* (pp. 61-75). Elsevier.

Hamidreza, Y. P., & Mohammadreza, H. A. (2016). Internet of Things (IoT): Applications, technologies and discussed challenges [In Persian].

Hashembeigi, F., Moshiri, F., & Asosheh, A. (2023). Introducing and Evaluating an Architectural Model for Smart Tourism Health Insurance based on IoT and Cloud.

Huang, X., Fan, W., Chen, Q., & Zhang, J. (2020). Energy-efficient resource allocation in fog computing networks with the candidate mechanism. *IEEE Internet of Things Journal*, *7*(9), 8502-8512.

Karimi, T., Azar, A., Mohebban, B., & Ghasemi, R. (2022). Developing an Internet of Things-based intelligent transportation technology roadmap in the food cold supply chain. *Industrial Management Journal*, *14*(2), 195-219.

Kazemi, S., & Saberi, S. (2022). The Role of Internet of Things (IoT) Technologies in Optimizing the Supply Chain of Startups.

Khadem, M., Toloie Eshlaghy, A., & Fathi Hafshejani, K. (2023). A novel elite-oriented meta-heuristic algorithm: qashqai optimization algorithm (QOA). *Journal of Information Systems and Telecommunication (JIST)*, *2*(42), 149.

Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2014). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, *103*(1), 14-76.

Mahmoudi, Z., Darbanian, E., & Nickray, M. (2021). Computational resource allocation in iot fog computing using teaching–learning-based optimization algorithm. *Journal of Soft Computing and Information Technology*, *10*(3), 73-85.

Mir, M., Yaghoobi, M., & Kheirabadi, M. (2023). Energy Aware Routing in the Internet of Things using improved Grasshopper Metaheuristic Algorithm. *Electronic and Cyber Defense*, *11*(1), 15-29.

Mirtaheri, S. L. (2023). Improving IoT resource management using fog calculations and ant lion optimization algorithm. *Journal of Information and Communication Technology*, *57*(57), 237.

Mohaghar, A., Sadeghi Moghadam, M. R., Ghourchi Beigi, R., & Ghasemi, R. (2021). IoT-based services in banking industry using a business continuity management approach. *Journal of Information Technology Management*, *13*(4), 16-38.

Mohammadi, M., & Ezadi Yegane, M. (2021). How can we equip academic libraries with IoT technologies: Practical guidelines. *International Journal of Digital Content Management*, *2*(2), 71-84.

Monfared, T., Nouroddin, S., & Ghatri, R. Z. (2023). Investigating the effect of IoT implementation on rail freight industry with a futures study approach. *Iranian Journal of Information and Communication Technology*, *55*(55), 191. [In Persian]

Saadi, H., Touhami, R., & Yagoub, M. C. (2020). Revolution of artificial intelligence and the internet of objects in the customer journey and the air sector. *Journal of Information Technology Management*, *12*(2), 59-69.

Sadegh, A. A., & Mohammadmehdi, M. (2023). Prioritization of IoT application areas in Iranian industries [In Persian].

Shahvaroughi Farahani, M., Nejad Falatouri Moghaddam, M., & Ramezani, A. (2023). Forecasting Tehran Price Index (TEPIX) Using Novel Meta-Heuristic Algorithms. *International Journal of Finance & Managerial Accounting*, *8*(28), 185-216.

Soleymani, M., & Nematzadeh, H. (2017). Scheduling of Real-time Processes Distribution on Multiprocessor Using Meta-Heuristic Ant Colony Algorithms, Genetics and PSO. *Journal of Advances in Computer Research*, *8*(3), 55-68.

Soltani, N., Barekatain, B., & SOLEIMANI, N. B. (2016). Job scheduling based on single and multi objective meta-heuristic algorithms in cloud computing: a survey.

Tabatabaei, S. (2023). Improving energy consumption in the Internet of Things using the Krill Herd optimization algorithm and mobile sink. *Journal of Information and Communication Technology*, *55*(55), 241.

Yeganeh, H., & Vaezpour, E. (2018). A Conceptual Framework and Architectural Considerations for Capability Enhancement in Software Defined Networks.

استناد به این مقاله: اعتدادی، علی. (۱۴۰۵). کارآمدی روش زمان‌بندی در پردازش لبه با نگاه به اینترنت اشیا و بلاکچین با استفاده از الگوریتم‌های فراابتکاری. *فصلنامه پژوهش‌های نوین در شهر هوشمند*، ۴(۳)، ۶-۲۱.

New Researches in The Smart City is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.