ستاد ملی هفته پژوهش و فناوری

دانشگاه بوعلی سینا
دانشکده مهندسی

ارائه‌ی مکانیسمی برای تعیین تابع توزیع بارکاری در پردازش‌های محاسبات مه بر پایه استفاده از سیستم‌های دسته‌بند یادگیر

گروه آموزشی کامپیوتر،
دانشکده مهندسی،
دانشگاه بوعلی‌سینا،
همدان

استاد راهنما: جناب آقای دکتر مهدی عباسی

دانشجو: بهاره حمیدی محب

رشته: معماری سیستم های کامپیوتری

مقطع: کارشناسی ارشد

b.hamidimoheb@eng.basu.ac.ir

چکیده

در سال های اخیر ، اینترنت اشیا (IoT) یکی از محبوب ترین فن آوری هایی است که تعاملات جدید میان افراد و انسان ها را برای ارتقا کیفیت زندگی تسهیل می کند. با توسعه سریع اینترنت اشیا ، الگوی محاسبات مه به عنوان یک راه حل جذاب برای پردازش داده های برنامه های اینترنت اشیا در حال ظهور است. در محیط مه ، برنامه های اینترنت اشیا توسط گره های محاسباتی میانی در مه و همچنین سرورهای فیزیکی در مراکز داده ابری اجرا می شوند. از طرف دیگر ، به دلیل محدودیت های منابع و انرژی ضروری است که مسائل مربوط به تصمیم گیری میزان بارکاری قابل پردازش به عنوان یکی از مشکلات چالش برانگیز در چشم انداز مه مورد توجه قرار گیرد. در این پژوهش قصد داریم راه‌حلی کارآمد و قابل پیاده‌سازی برای توزیع بارکاری مابین منابع پردازشی در لبه‌ی شبکه با در نظر گرفتن معیارهای انرژی و تأخیر و به شکلی بهینه ارائه دهیم. به طوری که از حداکثر توان منابع موجود در لبه ی شبکه بهره مند شویم. این مکانیزم بر پایه ی یادگیری تقویتی و سیستم های دسته بند یادگیر، با استفاده از الگوریتم XCSF تابعی را به عنوان میزان بارکاری پردازشی هر گره در لبه ی شبکه تقریب میزند.

کلمات کلیدی: اینترنت اشیاء، پردازش لبه، پردازش مه، یادگیری ماشین، تخمین تابع

مقدمه

- اینترنت اشیاء به ما اجازه می‌دهد، میلیاردها اشیاء فیزیکی را برای جمع‌آوری و مبادله داده‌ها به‌منظور ارائه برنامه‌های کاربردی، به یکدیگر متصل کنیم. نمونه‌ی این کاربردها، نظارت بر محیط‌زیست، مدیریت زیرساخت و دستگاه‌های خانگی به‌صورت خودکار است [۱]. اینترنت اشیا می‌تواند انسان و رایانه‌ها را برای یادگیری و تعامل با میلیاردها شیء که شامل حس‌گرها، ابزارها، سرویس‌ها و دیگر اشیاء متصل به اینترنت است در یک ارتباط شبکه‌ای فراهم کند. تحقق سیستم‌های اینترنت اشیاء، سبب ادغام یکپارچه دنیای سایبری با دنیای فیزیکی می‌شود و اساساً تغییر و تقویت تعاملات انسانی با جهان را ایجاد می‌کند [۲].

اینترنت اشیا

محاسبات ابری

محاسبات مه

سیستم‌های دسته بندی کننده یادگیر

مقدمه

اینترنت اشیا

محاسبات ابری

محاسبات مه

سیستم‌های دسته بندی کننده یادگیر

- در حال حاضر، داده‌ها از شبکه‌های اینترنت اشیاء که شامل وسایل هوشمند هستند، جمع‌آوری و متراکم شده و سپس به کارگزارهای ابری ارسال می‌شوند تا ذخیره‌سازی و مورد پردازش قرار گیرند. محاسبات ابری راه‌حلی را در سطح زیرساخت ارائه می‌دهد که پردازش داده‌های بزرگ را پشتیبانی می‌کند.
- در طول دهه گذشته، انجام محاسبات، کنترل و ذخیره‌سازی اطلاعات درشبکه‌های ابری، به مورد توجه قرارگرفته است. به‌طور خاص، محاسبات، ذخیره‌سازی و مدیریت شبکه به مراکز داده‌ی متمرکز، منتقل‌شده‌اند. بااین‌حال، با ظهور اینترنت اشیاء، محاسبات ابر با چالش‌های رو به رشدی مواجه است. از چالش‌های به وجود آمده می‌توان به تأخیرها، محدودیت پهنای باند، منابع محدود دستگاه‌ها، اهمیت ارتباط امن و بدون وقفه اشاره کرد [۳, ۴].

مقدمه

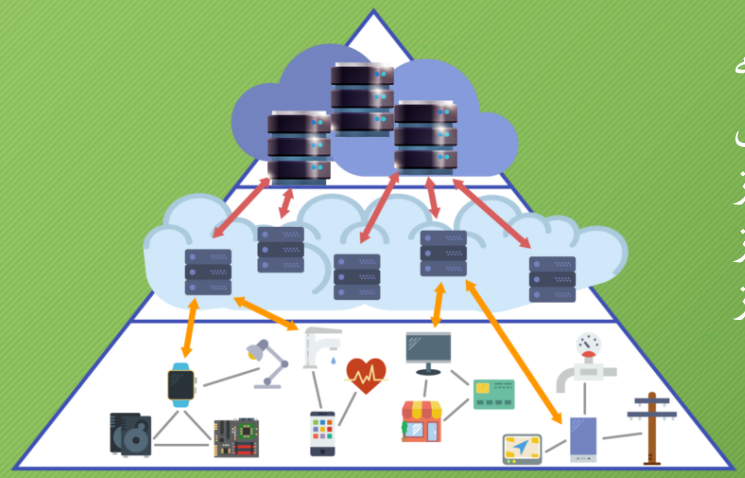
اینترنت اشیا

محاسبات ابری

محاسبات مه

سیستم‌های دسته بندی کننده یادگیر

تحقیقات نشان داده است که برای کاهش این مشکلات، به‌جای استفاده از ابرهای دور و متمرکز، از منابع غیرمتمرکز در لبه‌های شبکه برای پردازش داده‌های نزدیک به کاربر می‌توان استفاده کرد. این موضوع به عنوان محاسبات مه‌آلود/ لبه نامیده می‌شود [۵, ۶]. درنتیجه فنّاوری اینترنت اشیا، نیاز به یک معماری جدید دارد تا محاسبات، کنترل، ذخیره‌سازی و توابع شبکه را به دستگاه‌های کاربر نهایی نزدیک کند. پردازش مه‌آلود به‌صورت توزیع‌شده است و کنترل و مدیریت سیستم‌های پردازش مه‌آلود نیاز به متخصص ندارد [۳].



همانطور که در شکل مشاهده میشود، معماری محاسبات مه‌آلود از سه لایه تشکیل میشود : الف) لایه پایین : شامل حس‌گرها، اشیاء و سایر ابزار متصل به اینترنت؛ ب) لایه میانی: شامل گره‌های مه/لبه و ج) لایه بالایی: که مراکز داده و ابرها را ساختار آن را تشکیل میدهد. لایه میانی(گره‌های لبه) شبکه از ارسال غیرضروری اطلاعات به سمت مراکز داده جلوگیری میکند و برخی از وظایف ابر را به عهده میگیرد [۵].

مقدمه

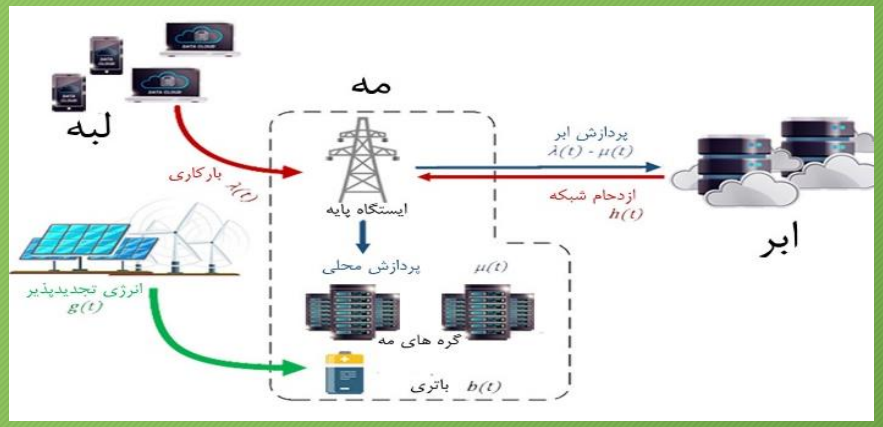
اینترنت اشیا

محاسبات ابری

محاسبات مه

سیستم‌های دسته بندی کننده یادگیر

- تخصیص بار، کار آسانی نیست زیرا معیارهای تخصیص منابع (که در بحث فوق ذکر شد) بسیار نزدیک و وابسته به یکدیگرند. به‌عنوان‌مثال، با توجه به محدودیت انرژی در لایه‌های پایین، بارکاری باید به مرکز داده‌ها فرستاده شود. از طرفی با ارسال این بارکاری به لایه‌های بالاتر، تأخیر به‌طور ناگهانی افزایش میابد. پس باید معیارهایی برای بارهای کاری مختلف تعیین کرده، سپس آن‌ها را اولویت‌بندی کرد تا بتوان یک استراتژی تخصیص متناسب با معیارها، انتخاب کرد. بنابراین سه معیار تأخیر، انرژی و هزینه در استفاده از منابع توسط بارهای کاری هم‌زمان، باید مدنظر قرارگرفته شود [۷]. از طرفی تخصیص بار در گره‌های لبه‌ی شبکه باید به صورتی باشد که از منابع موجود در لبه‌ی شبکه حداکثر بهره‌وری را داشته باشیم. از روش های یادگیری تقویتی می توان برای به حداقل رساندن زمان بیکاری گره‌های لبه‌ی شبکه و استفاده بهینه از منابع و خدمات مه استفاده کرد [۸]. در شکل زیر پردازش بارکاری را در لایه ی مه را نشان میدهد.



مقدمه

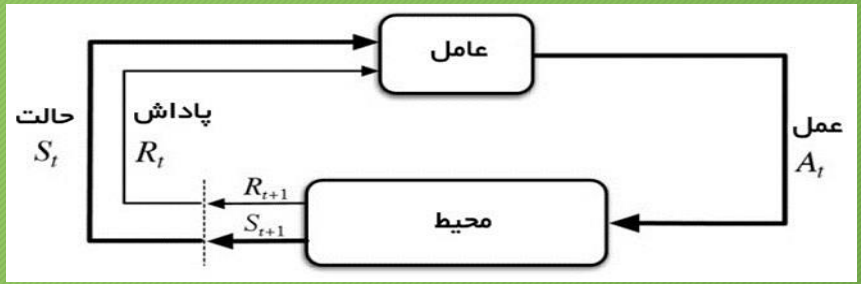
اینترنت اشیا

محاسبات ابری

محاسبات مه

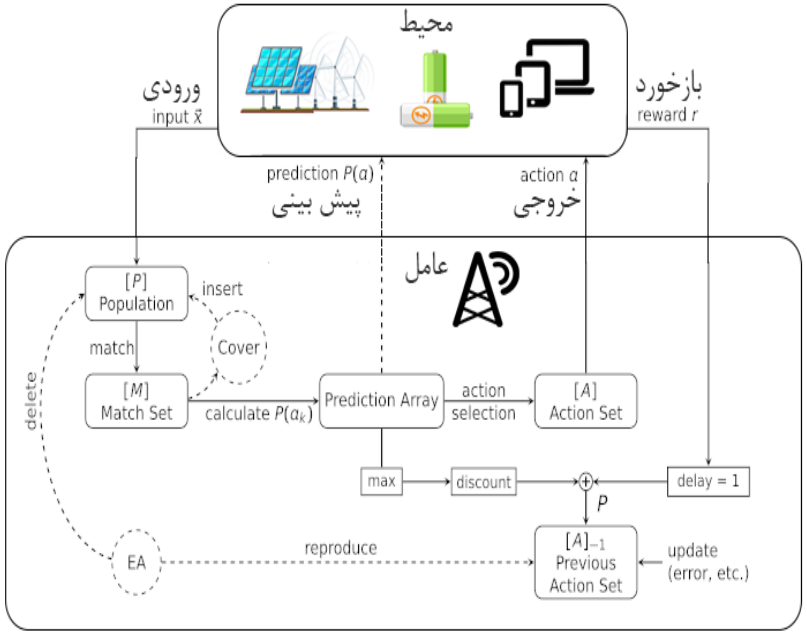
سیستم‌های دسته بندی کننده یادگیر

- در زمینه های یادگیری تقویتی ، به طور معمول از سیستم های دسته بند استفاده می شود. بنابراین تقریبی از تابع بازخورد خواهیم داشت که به سیستم اجازه میدهد برای هر ورودی که شامل میزان کل بار ورودی، وضعیت ازدحام شبکه و میزان انرژی تجدیدپذیر دریافت شده است، بهترین و بالاترین بازخورد را انتخاب کند [۹].
- سیستم‌های دسته‌بندی کننده یادگیر ، سیستمهایی مبتنی بر قانون هستند. در این سیستم‌ها قوانین عمدتاً به فرم " IF condition THEN action"هستند. از یک الگوریتم تکاملی یا روش‌های شهودی میتوان برای جستجو در فضای قوانین موجود و در همان زمان از فرآیند یادگیری دیگر برای انتساب نحوه استفاده به قوانین موجود میتوان استفاده نمود که درواقع باعث هدایت جستجو به سمت قوانین بهتر میشود [۱۰]. در محاسبات مه‌آلود، از سیستم دسته‌بندی کننده یادگیر برای کمک به تصمیم‌گیری ایستگاه پایه جهت تعیین میزان بار پردازشی در لبه شبکه استفاده میشود. شکل زیر نمایی از یک سیستم یادگیری تقویتی را که حوزه محاسبات مه آلود استفاده شده است نمایش می‌دهد. درواقع، ورودی این سیستم شامل میزان کل بار ورودی، وضعیت ازدحام شبکه و میزان انرژی تجدیدپذیر دریافت شده است و بازخورد همان هزینه ایست که به سیستم تحمیل شده است. الگوریتم مبتنی بر یادگیری نیز با هدف کاهش این هزینه‌ها عمل خواهد کرد[۱۱].



روش انجام تحقیق

در این پژوهش، به منظور بهینه‌تر کردن این روش توزیع بار را به عنوان تابع در هر گره در نظر می‌گیریم و بنابراین میزان بارکاری قابل پردازش در هر گره، باتوجه به ورودی سیستم، مقدار پویایی دارد به شکلی متعادل تر در هر گره توزیع خواهد شد و همچنین منجربه مدیریت منابع بهتر و کارایی بالاتر محاسبات مه‌آلود خواهد شد. بنابراین از یک سیستم دسته‌بند یادگیر به نام XCSF به منظور تعیین تابعی از میزان بارکاری پردازشی هر گره بهره‌مند خواهیم شد.



توزیع بار را به عنوان تابع در هر گره در نظر می‌گیریم و بنابراین از یک سیستم دسته‌بند یادگیر به نام XCSFبه منظور تعیین تابعی از میزان بارکاری پردازشی هر گره بهره‌مند خواهیم شد. برای این منظور از کتابخانه‌ی XCSF[۱۲] به زبان C++برای شبیه سازی استفاده خواهد شد. در شکل روبرو نمای کلی از XCSFمشاهده میشود. در این الگوریتم یک ورودی xرا از محیط دریافت میکند و یک خروجی را محیط میدهد و متناسب با آن یک بازخورد دریافت میکند که به بهبود عملیات یادگیری دسته بندها کمک میکند.

روش انجام تحقیق

تفاوت XCS و XCSF تنها در خروجی است. بازخورد و ورودی های آن یکسان است. XCS مقدار ثابتی را به عنوان خروجی میدهد درحالی که در XCSF خروجی به شکل تابع f(x) است. در این تابع w بردار وزنی است که در الگوریتم یادگیری تقویتی یادگرفته میشود. همانطور که قبلا هم ذکر شده بود، ورودی در محاسبات مه شامل میزان کل بار ورودی ، ازدحام شبکه و میزان انرژی تجدیدپذیر دریافت شده می‌باشد. بنابراین بردار ورودی ما شامل این سه مورد است و در هر گره پردازشی مه این تابع مشخص می‌شود. در واقع باید ضریب هر یک از این ورودی ها توسط سیستم‌های دسته-بند یادگیر تعیین شود.

	XCS در محاسبات مه	XCSF در محاسبات مه
ورودی (condition)	λ(t) : میزان کل بار ورودی h(t) : وضعیت ازدحام شبکه g(t): میزان انرژی تجدیدپذیر دریافت شده	λ(t) : میزان کل بار ورودی h(t) : وضعیت ازدحام شبکه g(t): میزان انرژی تجدیدپذیر دریافت شده
بازخورد(reward)	توان تاخیر مصرف باتری	توان تاخیر مصرف باتری
خروجی (action)	μ(t) : میزان بار قابل پردازش در لبه ی شبکه	f(x)= w . x´

$w = (w_0 , w_1 , \dots , w_n)$

$x´ = (x_0 , x_1 , \dots , x_n)$

فرضیه‌ها

- در تامین انرژی به دلیل **افزایش حجم پردازش‌ها در لبه‌ی شبکه و نبود منبع تولید انرژی در گره‌ها** ، با چالش مواجه هستیم.
- **قرار دادن منابع تولید انرژی** در لبه‌های شبکه، بسیار پرهزینه است.
- با برخورداری از **حداکثر توان منابع پردازشی موجود در شبکه** میتوان تا حدی این کمبود انرژی را جبران کرد.
- امکان ارائه **مدل جامعی از تخصیص بار** بین گره‌های پردازش مه‌آلود با در نظر گرفتن معیارهای ارزیابی کارایی توزیع وجود دارد.
- الگوریتم‌های موجود در حوزه بهینه‌سازی امکان بهینه‌سازی مدل ارائه‌شده را با درنظرگرفتن **معیارهای هزینه و انرژی** فراهم می‌کنند.
- بکارگیری الگوریتم های **یادگیری تقویتی و دسته‌بند یادگیر** در تخصیص بارها به شکلی پویا و متناسب با منابع موجود در لبه‌ی شبکه
- برای هر گره تابعی به فرم $f(x)= w \cdot x'$ ارائه خواهد شد. وزن ها(w) با کمک الگوریتم XCSF یادگیری خواهند شد تا بهترین تابع برای هر گره در نظر گرفته شود.

بحث و نتیجه‌گیری

ازآنجاکه برخی از جریان‌های کاری حساسیت زیادی به تأخیر در زمان پاسخ دارند، نمیتوان آن‌ها را به سمت ابرها فرستاد و به‌ناچار باید این محاسبات در گره‌های لبه شبکه صورت پذیرد. پس با افزایش حجم پردازش‌ها در لبه‌ی شبکه و نبود منبع تولید انرژی در این گره‌ها، محاسبات مه‌آلود برای تأمین انرژی، دچار چالش جدی میشود. از طرفی برای قرار دادن منابع تولید انرژی در لبه‌های شبکه، هزینه‌های زیادی باید متحمل شد تا انرژی موردنیاز برای این محاسبات تأمین گردد.

همانطور که قبلاً هم گفته شد ، محاسبات مه یک پلت فرم محاسباتی توزیع شده است که به جای انتقال از راه دور به ابر ، قابلیت پردازش و ذخیره اطلاعات را برای انجام خدمات اینترنت اشیا به صورت محلی فراهم می کند. مشابه رایانش ابری ، منابع ذخیره سازی ، محاسبات و شبکه را فراهم می کند. با توجه به پویایی ، ناهمگنی و عدم اطمینان محیط مه ، یک مکانیسم مدیریت منابع برای واقعیت بخشیدن محاسبات مه بسیار ضروری است [۱۳].

با این روش پیشنهادی ضمن بهینه سازی تعادل بین تاخیر و مصرف انرژی، از حداکثر توان منابع پردازشی موجود در لبه‌ی شبکه بهره‌وری میشود. از آنجایی که میزان بار در گره های لبه به صورت تابعی از ورودی‌های گره‌های لبه‌ی شبکه در نظر گرفته میشود؛ لذا هر گره در لبه‌ی شبکه میتواند مقدار بارکاری متفاوتی با توجه به منابع پردازشی موجود در گره، پردازش کند و همین امر موجب کارایی بالاتر در محاسبات مه‌آلود میشود. لذا با استفاده از حداکثر توان منابع پردازشی موجود در لبه‌ی شبکه این تعادل بین تاخیر و مصرف انرژی را بهینه‌تر می‌شود.

تقدیر و تشکر

از جناب آقای دکتر مهدی عباسی که راهنمایی این پایان نامه را به عهده دارند، کمال تشکر را دارم.

منابع

[1] J. Ni, K. Zhang, . X. Lin and . X. . S. Shen, "Securing fog computing for internet of things applications: Challenges and solutions," IEEE Communications Surveys & Tutorials, vol. 20, pp. 601-628, 2017.

[2] A. H. Ngu, M. Gutierrez, . V. Metsi, . S. Nepal and Q. . Z. Sheng, "IoT middleware: A survey on issues and enabling technologies," IEEE Internet of Things Journal, vol. 4, pp. 1-20, 2017.

[3] M. Chiang and . T. Zhang, "Fog and IoT: An overview of research opportunities," IEEE Internet of Things Journal, vol. 3, pp. 854-864, 2016.

[4] A. Al-Fuqaha, M. Guizani, M. Mohammadi, . M. Aledhari and . M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," IEEE Communications Surveys & Tutorials, vol. 17, pp. 2347-2376, 2015.

[5] C. -H. Hong and . B. Varghese, "Resource Management in Fog/Edge Computing: A Survey," arXiv preprint arXiv:1810.00305, 2018.

[6] P. Zhang, . M. Zhou and . G. Fortino, "Security and trust issues in Fog computing: A survey," Future Generation Computer Systems, vol. 88, pp. 16-27, 2018.

[7] W. Shi, J. Cao, Q. Zhang, Y. Li and . L. Xu, "Edge computing: Vision and challenges," IEEE Internet of Things Journal, vol. 3, pp. 637-646, 2016.

منابع

[8] F. M. Talaat, . M. S. Saraya, . A. I. Saleh, H. A. Ali and S. H. Ali , "A load balancing and optimization strategy (LBOS) using reinforcement learning in fog computing environment," Ambient Intell Human Comput, vol. 11, p. 4951–4966, 2020.

[9] S. W. Wilson, "Classifiers that approximate functions," Natural Computing, vol. 1, p. 211–234, 2002.

[10] M. . V. Butz, "Learning Classifier Systems".

[11] . M. Abbasi, M. Yaghoobikia, M. Rafiee, A. Jolfaei and M. R. Khosravi, "Efficient resource management and workload allocation in fog-cloud computing paradigm in IoT using learning classifier systems," Computer Communications, vol. 153, pp. 217-228, 2020.

[12] fchauvel, "XCSF," [Online]. Available: https://github.com/fchauvel/XSCF.

[13] M. Ghobaei-Arani, A. Souri and A. A. Rahmanian, "Resource Management Approaches in Fog Computing: a Comprehensive Review," Journal of Grid Computing, vol. 18, p. 1–42, 2020.