



ارائه شده توسط:

سایت ترجمه فا

مرجع جدیدترین مقالات ترجمه شده

از نشریات معتبر

بهینه سازی برنامه در رایانش ابری موبایل: انگیزش، رده بندی و چالش

های حل نشده

چکیده :

در رایانش ابری موبایل (MCC)، مهاجرت یک فرایند برنامه به مراکز داده های ابری امکان اجرای برنامه های منبع محور را در دستگاه های موبایل می دهد. با این حال، رویکرد های مهاجرت منبع محور و محدودیت های ذاتی محیط بی سیم مانع از دست یابی به عملکرد بهینه برنامه ها شده است. از این روی اجرای یک برنامه با هزینه پایین، سر بار حداقل و مهاجرت غیر نفوذی یک زمینه تحقیقاتی چالش بر انگیز می باشد. این مقاله یک سری چارچوب های اجرای برنامه موبایل پیشرفته را ارائه کرده و به بحث در خصوص راهبرد های بهینه سازی به منظور تسهیل دست یابی به طرح موثر، استفاده موثر و مهاجرت برنامه با عملکرد بهینه در MCC می می پردازد. ما اهمیت بهینه سازی عملکرد برنامه را با ارائه سناریو های زندگی واقعی که نیازمند طراحی موثر، استفاده کارآمد و اجرای بهینه برنامه در MCC است برجسته تر می کنیم. هم چنین این مقاله یک سری رده بندی های مربوط به برنامه موبایل مبتنی بر ابر را ارائه می کند. با این حال، ما چارچوب های اجرای عملکرد و دستگاه های موبایل در MCC را مقایسه می کنیم. بحث ما در خصوص دستور العمل های آینده تحقیقاتی برای بهینه سازی برنامه در MCC است. در نهایت، ما به برجسته سازی اهمیت کلیدی و دستور العمل های آینده تحقیقاتی در بهینه سازی برنامه موبایل مبتنی بر ابر می پردازیم.

لغات کلیدی: رایانش ابری موبایل، برنامه های موبایل مبتنی بر رایانش، بهینه سازی برنامه، اجرای برنامه موبایل

مبتنی بر ابر، چارچوب ها

فهرست موضوعات

1. مقدمه..... 53
2. سابقه و هدف..... 54
 - 2.1. رایانش ابری موبایل..... 54
 - 2.2. نرم افزار موبایل مبتنی MCC..... 54.

2.3.	بهینه سازی اجرای نرم افزار.	54
3.	انگیزه.	54
3.1.	پردازش تصویر	54
3.2.	تشخیص و ترجمه صدا.	55
3.3.	گیمینگ M.	55
3.4.	سنجش طیفی مشارکتی در شبکه های ابر رادیویی شناختی.	55
4.	چارچوب های اجرای برنامه موبایل مبتنی بر ابر پیشرفته.	55
4.1.	چارچوب های مبتنی بر بهینه سازی تک هدفه.	55
4.2.	چارچوب های مبتنی بر بهینه سازی دو هدفه.	57
4.3.	چارچوب های مبتنی بر بهینه سازی چند هدفه.	60
5.	رده بندی های برنامه محور موبایل رایانش محور.	62
5.1.	رده بندی راهبرد های بهینه سا	62
5.2.	رده بندی چارچوب های اجرای برنامه در MCC	62
5.3.	طبقه بندی چارچوب اجرای نرم افزار.	64
6.	شاخصی برای برنامه مهاجرت مطلوب MCC	64
6.1.	دستگاه موبایل	64
6.2.	نوع برنامه	65
6.3.	تنظیمات کاربر	65
6.4.	هزینه	66
6.5.	شبکه	66
7.	جهت تحقیقات آینده.	66
8.	چالش باقی مانده.	66
8.1.	نرم افزار بهینه و طراحی چارچوب اجرا.	67

- 8.2. استقرار کارآمد و اجرای کاربر شفاف. 67
- 8.3. مدیریت بهینه سازی شده بیدرنگ از محیط محاسبات ناهمگن. 67
- 8.4. تأمین خدمات خودکار. 67
- 8.5. مقیاس پذیری. 67
- 8.6. در دسترس بودن خدمات. 67
9. نتیجه گیری. 67
- سپاسگزاری. 67
- منابع. 67

1- مقدمه

در طی دهه اخیر، پیشرفت های فناوری های شبکه و رشد دستگاه های محاسباتی و رایانشی موجب تحقق رویای رایانش فرا گیر شده است. پیشرفت های قابل توجه و مطلوب، یک نیروی محرک را برای تعدادی از برنامه های الکترونیک نظیر یادگیری الکترونیک، تجارت الکترونیک، دستور العمل الکترونیک، بهداشت الکترونیک و بازی های اینترنتی ارائه کرده اند. اخیراً، با ظهور فناوری های بی سیم (چین و همکاران 2014، احمد و همکاران 2015 ب،ج)، و دستگاه های موبایل، پارادایم برنامه الکترونیک به سمت پارادایم برنامه های m تغییر یافته است. از این روی، با ظهور طیف وسیعی از برنامه های موبایل، نظیر یادگیری موبایل (متیولالا 2007)، سلامت موبایل (کروز و باروس 2005)، دستور العمل های موبایل (اپرمن و اسپشت 1999)، بازی های موبایل (بالاگاس و همکاران 2007) و برنامه های کارگر سیار (مازولنی و تای 2007)، بخشی از مجموعه ای از برنامه های کاربر سیار است. کاربران موبایل انتظار می رود که برنامه های m را با سطح عملکرد مشابه در سیستم های کامپیوتری ساکن استفاده و اجرا کنند. با این حال دستگاه های موبایل دستگاه های با محدودیت منابع هستند که سطح یکسانی از تجربه کاربری را فراهم نمی کنند.

در این رابطه، مهاجرت رایانشی به صورت یک راه حل سطح نرم افزاری مهم در نظر گرفته می شود که موجب کاهش محدودیت منابع دستگاه های موبایل با مهاجرت برنامه ها به کامپیوتر های ساکن قابل دسترس می شوند)

هیورت-کانپ و لی 2010، مارلینلی 2009، کویال و کارتر 2004، کورو و همکاران 2010، کووچ و همکاران 2012 ب، وربلان و همکاران 2012 ب، فرشیان و همکاران 2012، یانگ و همکاران 2008، چان و مانیتاس 2009). مهاجرت رایانشی توسط چارچوب های اجرای برنامه مدیریت می شود. مشابه با دیگر زمینه های تحقیقاتی نظیر شبکه های ارتباطی (شامشیرباند و همکاران 2015، زنگ 2008) و سیستم های توزیعی نظیر رایانش ابری (نینگ و همکاران 2013)، روش های بهینه سازی در چارچوب های اجرای برنامه MCC استفاده می شوند. چارچوب های اجرایی، توابع بهینه سازی هدف را به صورت زیر در نظر می گیرند: ذخیره برق پردازشی، استفاده از پهنای باند کارآمد و کمینه سازی مصرف انرژی. به طور خلاصه، چارچوب ها برای بهینه سازی هزینه اجرایی طراحی می شوند. هدف کلی این رویکرد ها، عملی کردن برنامه های موبایل بر روی دستگاه های موبایل با محدودیت منابع می باشد. اجرای اجزای رایانشی برنامه موبایل در رایانش ابری سیار MCC مستلزم تقسیم برنامه پیچیده در سطوح مختلف و مهاجرت اجزا به گره سرور شبکه می باشد (احمد و همکاران 2015 الف). این مکانیسم های منبع محور و القا کننده تاخیر به شدت بر تجربه کاربر اثر منفی می گذارند. از این روی، استفاده از روش های سبک وزن برای اجرای بهینه برنامه های موبایل در MCC ضروری است. اجرای بهینه اشاره به وضعیت اجرای برنامه در MCC دارد که عملکرد بهینه نسبت به اجرای محلی با حداقل هزینه و سربار پایین دارد. طراحی موثر برنامه ها موجب کاهش هزینه، افزایش قابلیت اطمینان و عملکرد عالی می شوند. استفاده کارآمد موجب اطمینان از این می شود که اجزای برنامه های موبایل در MCC دارای وابستگی حداقل به یک دیگر می باشند. این موجب کاهش هزینه اجرا و سربار عملیاتی می شود.

اگرچه چندین مقاله (ابولفضیلی و همکاران 2013، فرناندو و همکاران 2013، دینه و همکاران 2013، کامور و همکاران 2013، خان و همکاران 2014، رحیمی و همکاران 2014) به بررسی ابعاد مختلف اهرم بندی سرویس های ابری برای افزایش قابلیت های دستگاه های موبایل پرداخته اند، بهینه سازی اجرای برنامه در MCC هنوز مطالعه نشده است. در مقاله ابولفضیلی و همکاران 2013، ما به طور جامع به بررسی تقویت موبایل مبتنی بر ابر و روش های مختلف برای افزایش پتانسیل دستگاه های موبایل می پردازیم. کار های تحقیقاتی انجام شده توسط فرناندو و همکاران (2013)، دین و همکاران (2013) و رحیمی و همکاران (2014)، یک نظر سنجی و مطالعه جامع را بر روی MCC ارائه کرده است که کاربرد، معماری، مسائل و چالش ها را در بر می گیرد. با این حال، این

مقاله اولین مطالعه ای است که به بررسی چارچوب های اجرایی برنامه پیشرفته برای شناسایی رویکرد های بهینه سازی مورد استفاده توسط طراحان، طبقه بندی رویکرد های بهینه سازی و برجسته سازی چالش های دخیل در دست یابی به بهینه سازی برنامه می پردازد.

اهداف این مقاله شامل 1- بررسی چارچوب های اجرایی برنامه پیشرفته در MCC، 2- شناسایی رویکرد های بهینه سازی مربوط به طراحی، استفاده و اجرای برنامه در MCC، 3- طبقه بندی و ارزیابی رویکرد های شناسایی شده در شکل رده بندی 4- شناسایی چالش های تحقیقاتی اشکار در بهینه سازی طراحی، استفاده و اجرای برنامه در MCC است. مقایسه نشان دهنده نقاط مشترک و نقاط تفاوت در میان چارچوب های اجرایی پیشرفته بر اساس پارامتر معنی دار است که عملکرد کاربرد اثر دارد. برخی از پارامتر ها، شامل تاخیر در انتقال، پشتیبانی کیفیت خدمات، سربار پروفیل، مقیاس پذیری و هزینه عملیاتی هستند. ما هم چنین شاخص های مهاجرت برنامه را ارزیابی می دهیم که در تصمیم مهاجرت بهینه سازی می شود. در نهایت، ما در مورد تحقیقات آینده مورد نیاز برای بهینه سازی طراحی کاربرد، استفاده و اجرا در MCC بحث می کنیم.

این مقاله به بخش های زیر سازمان دهی می شود. بخش 2 مفاهیم اساسی MCC، برنامه موبایل مبتنی بر ابر و اجرای آن در MCC و بهینه سازی اجرای برنامه را معرفی می کند. بخش سوم به بحث در خصوص سناریو هایی می پردازد که نشان دهنده ملزومات بهینه سازی برنامه در MCC است. بخش چهارم شامل مرور منابع برای اجرای برنامه برای MCC بوده و مطالعه ای تفصیلی از چارچوب های اجرایی برای برجسته سازی نقاط قوت و ضعف چارچوب های موجود ارزیابی می کند. رده بندی های برنامه محور در بخش 5 ارزیابی شده است که موارد زیر را پوشش می دهد 1- راهبرد های بهینه سازی برنامه 2- عملیات اجرای برنامه و 3- چارچوب های اجرای برنامه. در بخش 6، مناسبیت شاخص های مختلف برای کمک به طراحان چارچوب در انتخاب شاخص مناسب برای دست یابی به عملکرد برنامه بهینه سازی شده بررسی می شود. مطالعات آینده بر اساس مرور منابع در بخش 7 است. در نهایت چالش های تحقیقاتی اشکار در بخش 8 و 9 ارزیابی می شود و سپس نتیجه گیری با ارزیابی خلاصه و اطلاعاتی برای مخاطبان به پایان می رسد.

2- پیش زمینه

این بخش یک پیش زمینه از MCC، برنامه موبایل مبتنی بر ابر و اجرای آن در MCC ارائه می کند. به علاوه، این بخش مرور مختصری بر بهینه سازی اجرای برنامه برای ارائه دانش بنیادین به مخاطبان دارد.

2-1 رایانش ابری موبایل

MCC جدید ترین مدل رایانشی توزیعی است که موجب توسعه خدمات و منابع ابر رایانشی برای کاهش محدودیت های منابع در خدمات موبایل می شود. MCC موجب کاهش توسعه و هزینه اجرایی برنامه های موبایل شده و از این روی به کاربر موبایل امکان دستیابی به فناوری جدید را بر اساس تقاضا می دهد. از این روی، این مدل، توجه شرکت ها را به صورت کسب و کار های سود آور جلب کرده است. MCC بر کاهش محدودیت های منابع در دستگاه های موبایل با استفاده از راهبرد های مختلف تاکید دارد: نظیر تشدید فیلتر، تشدید انرژی، تشدید ذخیره و تشدید پردازش برنامه دستگاه های موبایل (ابولفضلی و همکاران 2013). MCC از دستگاه های ذخیره ابر های رایانشی برای ذخیره برون دستگاهی استفاده کرده و به خدمات پردازش برنامه سرور ابر دسترسی دارد تا بتواند برنامه های رایانشی بر روی دستگاه های موبایل تولید کند. با این حال، یک دسترسی بهینه به فهرست ها در مراکز داده های ابر ذخیره سازی می شود که هنوز یک دیدگاه تحقیق چالش بر انگیز است.

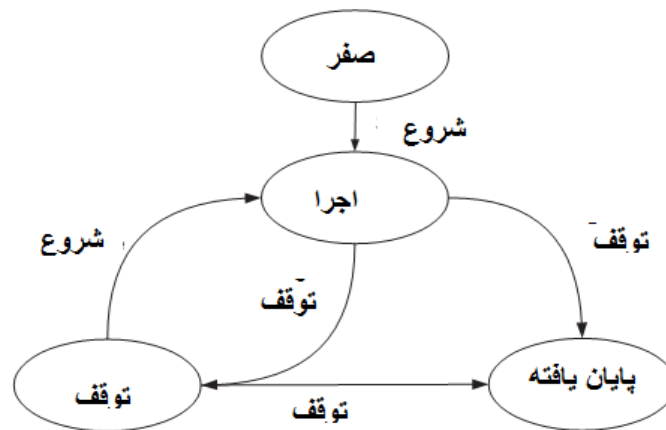
2-2 برنامه موبایل مبتنی بر ابر و اجرای آن در MCC

برنامه های موبایل مبتنی بر ابر می توانند بر روی دستگاه های موبایل و نیز ابر اجرا شوند. برنامه های موبایل مبتنی بر ابر متشکل از دو احزا می باشند که شامل حافظه محور و رایانش محور هستند و با سخت افزار های موبایل تعامل ندارند در حالی که اجزای غیر قابل انتقال برای کارکرد های خاص نظیر دسترسی به سخت افزار، تعامل کاربر و کار های مربوط به امنیت استفاده می شود (کوو و همکاران 2010). فرایند تقسیم اجزای برنامه به دو مجموعه، قابل انتقال و غیر قابل انتقال موسوم به تقسیم بندی است. معمولاً، تقسیم بندی به سه شیوه مختلف انجام می شود از نظر آماری، دینامیک، و نیمه دینامیک.

اجرای برنامه موبایل در MCC را می توان با نمودار وضعیت نشان داده شده در شکل 1 توضیح داد. اجرای برنامه زمانی شروع می شود که کاربر یک ایکون برنامه را کلیک می کند. برنامه زمانی به حالت اجرا وارد می شود که وظایف مختلفی را اجرا کند. برای مهاجرت برنامه به ابر، چارچوب اجرایی موجب توقف برنامه اجرایی می شود. برنامه زمانی در حالت اجرا قرار می گیرد که چارچوب اجرا موجب توقف شود. سپس کنترل اجرایی به سرور ابری

مهاجرت می کند که در آن برنامه از نو شروع شده و با استفاده از وضعیت های مطلوب پیکر بندی می شوند. با تکمیل اجرا بر روی سرور ابری، نتایج وارد دستگاه موبایل می شوند که در آن برنامه مجددا در دستگاه موبایل اجرا می شود. در نهایت، با تکمیل اجرا، برنامه متوقف شده و وارد یک حالت نهایی می شوند.

شکل 1: نمودار حالت اجرای برنامه (احمد و همکاران 2015 الف)



2-3 بهینه سازی اجرای برنامه

بهینه سازی اجرای برنامه برای MCC به دلیل ماهیت محدود بر منابع دستگاه های موبایل ، محیط اجرای توزیعی و محدودیت های ذاتی محیط ارتباطی مهم است. عملکرد اجرای برنامه در MCC را می توان با دست یابی به طراحی موثر برنامه، با استفاده موثر و با اجرای برنامه بهینه در محیط توزیع یافته بهبود بخشید. از این روی برنامه اجراء در MCC نیازمند طراحی موثر طراحی کارآمد و اجرای بهینه در MCC می باشد. طراحی موثر یک برنامه مستلزم این است که برنامه عملکرد عالی، کم هزینه و با قابلیت اطمینان بالا داشته باشد. استفاده کار مستلزم این است که اجزای برنامه های موبایل در MCC دارای حداقل وابستگی به یک دیگر باشند و این موجب کاهش هزینه های اجرایی و سربار عملیاتی می شوند. اجرای بهینه امکان اجرای برنامه پیوسته را با دخالت کاربر می دهد. چارچوب های اجرای برنامه با هدف بهینه سازی ماتریس برنامه طراحی می شوند. مثال ها شامل موارد زیر هستند: کمک موبایل با استفاده از زیر ساخت (MAUI) (گکرا و همکاران 2010) برای بهینه سازی مصرف انرژی دستگاه موبایل طراحی می شود.

3- انگیزش:

این بخش به ارزیابی سناریوهای برنامه در حوزه MCC برای ارزیابی هدفی برای انجام تحقیقات می پردازد. بهینه سازی اجرای برنامه در سناریوهای مختلف نظیر پردازش تصویر، تشخیص و ترجمه صدا، بازی های موبایل، سنجش طیفی مشارکتی در شبکه های ابر رادیویی شناختی به دلیل ماهیت منبع محور برنامه و محدودیت های دستگاه های موبایل و محیط ارتباطی انجام می شود.

3-1 پردازش تصویر

سناریوی بحث شده توسط هاترو-کانپا و لی (2010) یک مثال از برنامه موبایل پردازش تصویر بحرانی- زمانی رایانشی می باشد. در سناریوی نمونه " پیتر که یک بازدید کننده خارجی در کره جنوبی است، یک نمایشگاه جالب را با یکتوصیف کتبی در زبان کره پیدا کرده است" او زبان کره ای کتبی را نمی داند. او تصویری از نمایشگاه را دیده و می خواهد تا کل تصویر متن را با استفاده از برنامه تشخیص ترکیب پردازش کند. اجرای برنامه مستلزم محاسبه و رایانش بالاتری است به طوری که دستگاه موبایل منبع محور قادر به اجرای آن نیست. او می تواند نرم افزار را در سه شیوه متفاوت اجرا کند: در(الف) ابر دور،(ب) تکه ابر، و (ج) ابر مجازی یا ابر موقت همراه. برای سادگی ، ما در نظر می گیریم که او ابر موقت موبایل را برای اجرای نرم افزار انتخاب می کند. او ابزار های موبایل بازدید کننده های دیگری را که دارای هدف یکسانی برای ترجمه توصیف هستند مشاهده می کند. او این منابع دستگاه موبایل را با استفاده از منابع دستگاه موبایل سایر مسافران نزدیک تقویت می کند. سپس، ابر یک باره موبایل، برنامه را برای ترجمه زبان و استخراج متن اجرا می کند.

دستگاه های موبایل دارای منابع محدودی بوده و این دستگاه ها با باتری کار می کنند. از این روی، فرایند اجرای برنامه بایستی به طور کارآمد منابع موجود را مصرف کرده و توان باتری را حفظ کند. به علاوه، پیتر و سایر مسافران در موزه برای مدت کوتاهی بوده و آن ها باید تعداد توصیفات را استخراج کنند و این برای کمینه سازی زمان استخراج برای مسائل زمان واقعی مهم است. هم چنین اختلال ناشی از مهاجرت برنامه باید به حداقل برسد تا یک تجربه کاربری قابل قبول را فراهم کند. چارچوب های اجرای برنامه بهینه در این سناریو موجب افزایش قدرت کشف تعدادی از نمایشگاه ها در طی مسافرت کوتاه با سربار پایین دستگاه های موبایل می شود. هدف فرایند مهاجرت برنامه کاهش و کمینه سازی مصرف انرژی در طی مهاجرت است. با این حال یک سری چالش هایی در تحقق اجرای برنامه بهینه در این سناریو وجود دارد. کشف دستگاه برای ایجاد ابر موبایل، چالش مهمی

برای تحقق اجرای برنامه بهینه در این سناریو است. پارتیشن بندی برنامه، زمان بندی، مهاجرت برنامه، احراز هویت و اطمینان از حریم خصوصی داده های شخصی بر روی دستگاه موبایل، چالش های رایج اصلی در تحقق اجرای برنامه بهینه هستند.

3-2 تشخیص و ترجمه صدا

اهمیت درک و تحقق بهینه سازی اجرای برنامه در MCC می تواند با یک سناریویی برجسته سازی شود که در آن پیتر، به عنوان یک توریست، برخی از کره ای ها را در موزه دیده و از آن ه می خواهد تا در مورد نمایشگاه حرف بزنند. او زبان کره ای نمی داند و محلی ها هم زبان انگلیسی را نمی دانند. اکنون پیتر می خواهد تا از تشخیص و ترجمه صدا استفاده کند. نرم افزار برای اجرا بر روی دستگاه موبایل سنگین است و از این روی او به ابر وابسته است. او می تواند نرم افزار را به سه شیوه متفاوت در محیط MCC اجرا کند: الف: ابر موبایل ب: تکه ابر و ج: ابر دور. در این مورد، می توان در نظر گرفت که او ابر دور را برای اجرای نرم افزار انتخاب می کند. او باید تعدادی از وظایف را قبل از اجرای واقعی نرم افزار نظیر کشف سرور ابر، اسکن سرویس، احراز هویت و سایر کار ها برای تثبیت پلتفرم اجرای نرم افزار در MCC انجام دهد. همه این کار ها باید برای بهبود زمان اجرای برنامه و سربار بهینه سازی کرد. چون پیتر و کره ای ها در موزه برای مدت کوتاهی قادر به تحمل تاخیر در اجرای برنامه در ابر نمی باشند. به منظور حفظ استفاده از تشخیص صدا و نرم افزار ترجمه، زمان اجرا در ابر باید به حداقل برسد. دستگاه ای موبایل دستگاه های با محدودیت منابع هستند و از این روی، اجرای برنامه در MCC باید از منابع دستگاه کمینه استفاده کند.

3-3 بازی موبایل

بازی موبایل مبتنی بر ابر دیگر نمونه برای تحقق و درک نیاز به بهینه ساز اجرای برنامه در MCC می باشد. پیتر بازی را بر روی موبایل خود انجام می دهد به خصوص زمانی که منتظر خوردن نهار در رستوران است. به طور ناگهانی باطری موبایل کم شده و دستگاه می خواهد خاموش شود. این وضعیت منجر به اختلال اجرای برنامه بر روی دستگاه موبایل می شود. در عوض، اگر چارچوب اجرای مبتنی بر ابر بر روی دستگاه موبایل اجرا شود، پیتر قادر به افلود کردن بازی به سرور و یا به ابر دور است به خصوص زمانی که می خواهد تا بازی انجام دهد. افلودینگ یا تخلیه بازی به سرور نزدیک یا ابر موجب افزایش عمر باطری موبایل شده و پیتر می تواند بدون مشکل بازی

کند. با این حال تصمیم افلودینگ بر اساس توان باطری باقی مانده بازی در سرور موجود است. کشف سرور و سرویس های ارایه شده توسط سرور و احراز هویت آن ها مستلزم فرایند نیازمند محاسبه و زمان است و باید برای استفاده از منابع موجود نزدیک به شیوه ای شفاف برای کاربر بهینه سازی شود (احمد و همکاران 2013).

3-4- سنجش طیفی مشارکتی در شبکه های ابر رادیوی شناختی

در ووو همکاران (2012)، محققان دارای MCC تلفیقی با شبکه های رادیویی شناختی برای پردازش حجم زیادی از داده های سنجش طیفی برای تولید یک گزارش کامل برای تشخیص کاربر در نزدیکی است. ماهیت حساس به زمان سناریو نیازمند منابع محاسباتی بالا همراه با احراز هویت سریع برای شناسایی مقلدان و مکانیسم های یکپارچی داده های سبک وزن برای کنترل جعلی بودن داده ها می باشد. گزارش سنجش غیر صحیح منجر به هشدار کاذب و سوء انتخاب کاربر اصلی می شود. ماهیت حساس به زمان سنجش طیفی نیازمند اجرای بهینه الگوریتم سنجش مشارکتی در محیط MCC و ماهیت حساس به زمان نیازمند اجرای بهینه الگوریتم سنجش مشارکتی در محیط MCC و ماعیت دستگاه های موبایل نیازمند کاربرد موثر منابع است. محدودیت های محیط ارتباطی مستلزم کاهش سربار و هزینه ارتباطات است.

4- چارچوب های برنامه موبایل مبتنی بر ابر پیشرفته

چارچوب اجرای برنامه برای اجرا در محیط متنوع طراحی شده و نیاز های انواع مختلف کاربرد ها را برآورد می کند. هدف اصلی چارچوب های اجرای برنامه در MCC، تقویت منابع دستگاه های موبایل با اهرم بندی منابع و خدمات ابر است/ با این حال، تعداد پارامتر هی دخیل در بهینه سازی از یک چارچوب به چارچوب دیگر متغیر است. بر اساس تعداد پارامتر های بهینه سازی، ما چارچوب های اجرای برنامه موبایل مبتنی بر ابر پیشرفته را به سه مقوله طبقه بندی کرده ایم. چارچوب های مبتنی بر بهینه سازی یک منظوره، چارچوب های مبتنی بر بهینه سازی دو منظوره و چارچوب های مبتنی بر بهینه سازی چند منظوره.

4-1 چارچوب های مبتنی بر بهینه سازی تک منظوره

چارچوب های مبتنی بر بهینه سازی تک منظوره عمدتاً بر بهینه سازی یک هدف تاکید دارند. چارچوب های مبتنی بر بهینه سازی تک منظوره ساده ترین شکل از بهینه سازی است ولی مطابق با محیط اجرای متنوع نیست

و قادر به رفع نیاز های برنامه های مختلف نیست. در این جا ما چارچوب هایی را ارائه می کنیم که هدف آن بهینه سازی یک هدف است:

الف: ابر کلون: در مقاله چان و همکاران (2011)، ابر کلون، یک متخصص برنامه انعطاف پذیر، پیشنهاد می شود که موجب افزایش قدرت برنامه های موبایل برای افلود بخشی از آن به ابر دور می شود. این سیستم از تحلیل استاتیک و پروفیل بندی دینامیک برای پارتیشن بندی برنامه موبایل استفاده می کند. هدف اصلی پارتیشن بندی، بهینه سازی هزینه اجرای کل است. هزینه اجرا شامل هزینه محاسبه، $Comp(E)$ ، و هزینه مهاجرت $Migr(E)$ می باشد. هزینه مصرف انرژی متشکل از فعالیت CPU، وضعیت نمایش و وضعیت شبکه است:

$$C(E) = Comp(E) + Migr(E) \quad (1)$$

$$Comp(E) = \sum_{i \in E, m} [(1 - L(m))I(i, m)C_c(i, 0) + L(m)I(i, m)C_c(i, 1)] \quad (2)$$

هزینه محاسباتی مقداری را از متغیر های هزینه کلون $C_c(i;1)$ اختیار می کند به خصوص زمانی که روش بر روی کلون دستگاه موبایل اجرا می شود یا متغیر های هزینه دستگاه موبایل، $C_c(i;0)$ استفاده می شود. هزینه مهاجرت مجموع هزینه مهاجرت فردی $C_s(i)$ درخواست هایی است که روش های آن ها دارای نقاط مهاجرت هستند:

$$Migr(E) = \sum_{i \in E, m} R(m)I(i, m)C_s(i) \quad (3)$$

معادله (14) محدودیتی را مدل سازی می کند که دو روش به طور مستقیم هم دیگر را درخواست می کنند که در یک مکان قرار نمی گیرد. معادله 5 نشان می دهد که روش در نظر گرفته شده برای اجرا بر روی دستگاه موبایل تنها بر روی دستگاه موبایل اجرا می شود. معادله 6 اطمینان حاصل می کند که روش های وابسته به کلاس یکسان حالت بومی CS در یک مکان قرار گرفته است. معادله 7 اطمینان حاصل می کند که همه روش ها توسط روش مهاجرت شده انتخاب شده و نباید مهاجرت کنند:

$$L(m_1) \neq L(m_2), \forall m_1, m_2 : DC(m_1, m_2) = 1 \wedge R(m_2) = 1 \quad (4)$$

$$L(m) = 0, \forall m \in V_M \quad (5)$$

$$L(m_1) = L(m_2), \forall m_1, m_2, C : m_1, m_2 \in V_{Nat_C} \quad (6)$$

$$R(m_2) = 0, \forall m_1, m_2 : TC(m_1, m_2) = 1 \wedge R(m_1) = 1 \quad (7)$$

$R(m)$ یک متغیر تصمیم که نشان دهنده افزایش نقطه مهاجرت در روش m در ورودی و خروجی روش است به خصوص اگر $R(m)$ یک پارتیشن بندی باشد که نقطه مهاجرت را می افزاید در غیر این صورت روش اصلاح نشده باقی می ماند. رابطه بین دو روش با $DC(m_1, m_2)$ و $TC(m_1, m_2)$ نشان داده می شود که در آن $DC(m_1, m_2)$ به صورت زیر خوانده می شود: روش m_1 مستقیماً روش m_2 را درخواست می کند و $TC(m_1, m_2)$ به صورت زیر خوانده می شود: روش m_1 به طور موقت روش m_2 را درخواست می کند. $L(m)$ یک متغیر تصمیم می باشد که نشان دهنده موقعیت روش m است. V_M و V_{Nat_C} بیانگر مجموعه ای از روش ها و روش های کلاس C است.

ب: چارچوب پارتیشن بندی کاربرد شاخه جریان: یک چارچوبی برای پارتیشن بندی بهینه برنامه شاخه یا جریان داده ها برای بیشینه سازی سرعت و بازدهی در فرایند در مقاله یانگ و همکاران (2012) ارائه شده است. مسئله پارتیشن بندی به صورت مسئله تعیین و تخصیص مجموعه ای از اجزای گراف جریان داده به منابع با هدف بیشینه سازی بازدهی برنامه داده مدل سازی می شود. برای کاهش وزن چارچوب، روش بهینه ساز در طرف ابر باقی می ماند. تابع هدف به صورت زیر تعریف می شود

$$\begin{aligned} \max_{x_i, y_{ij}} TP &= \frac{1}{t_p}, \quad i, j \in \{0, 1, \dots, v+1\} \quad \text{where} \\ t_p &= \max \left\{ \max_{i \in V} \left(x_i \cdot \frac{S_i}{\eta p} \sum_{i \in V} x_i \right), \max_{(i,j) \in E} \left(\frac{d_{ij} (x_i - x_j)^2}{y_{ij}} \right) \right\} \\ \text{s.t.} & \begin{cases} \sum_{ij \in E} y_{ij} (x_i - x_j)^2 = B, \\ y_{ij} > 0, \\ x_0 = 1, \\ x_{v+1} = 1, \\ x_i = 0 \text{ or } 1, \quad i \in \{1, 2, \dots, v\} \end{cases} \quad (8) \end{aligned}$$

که η و P قابلیت سی پی یو دستگاه موبایل و درصد منابع CPU ایده آل دستگاه موبایل به ترتیب است. در معادله B.8 پهنای باند است. متغیر X_i ، متغیر تصمیم گیری ای است که می تواند مقدار 1 یا 0 را اختیار کند. $X_i=1$ به معنی اجرای جزء i بر روی دستگاه موبایل است در غیر این صورت $X_i=0$ بر روی ابر اجرا می شود. متغیر $y_{i,j}$ نشان دهنده پهنای باند بی سیم تخصیص داده شده به کانال (i,j) می باشد. دو گره مجازی 0 و $v+1$ برای کاهش محدودیت افزوده می شوند که داده ها از آن گرفته شده و به دستگاه موبایل تحویل داده می شوند. متغیر های $d_{0,1}$ و $d_{v,v+1}$ بیانگر یک پهنای باند بی سیم تخصیص داده شده به کانال (i,j) می باشد. دو گره مجازی 0 و $v+1$ برای کاهش محدودیتی که داده ها از آن ناشی می شود تخصیص داده می شوند. با این حال، الگوریتم پارتیشن بندی مبتنی بر ژنتیک مورد استفاده توسط چارچوب موجب اطمینان از زمان پاسخ بهینه سازی ثابت نمی شود.

پ: میان افزار دینامیک آلود رایانشی (CODM): در ورلن و همکاران (2011)، یک چارچوب میان افزار برای انتخاب استفاده از پیکر بندی با بهترین کیفیت با در نظر گرفتن منابع و ارتباطات موجود ارائه می شود. واحد اصلی استفاده، سبد یا دسته است. برای هر سبد، چارچوب، پیکر بندی های مختلف را با سطوح مختلف تقاضای منابع و سطوح کیفیت مختلف ارائه می شود

بهترین پیکر بندی با تابع هدف مدل سازی می شود

$$\begin{aligned} & \max_{\text{configurations}} \sum_i w_i \\ & \text{s.t.} \begin{cases} \forall m : \sum_i X_{im} \times w_i \leq M_m \\ \forall i : \sum_m X_{im} = 1 \end{cases} \end{aligned} \quad (9)$$

متغیر w_i بیانگر وزن i امین مولفه ای است که کیفیت مولفه را نشان می دهد. متغیر تصمیم X_{im} در صورتی برابر با 1 است که به دستگاه تخصیص داده شود در غیر این صورت 1 است. اولین محدود کننده این اطمینان را به ما می دهد که مجموع اوزان همه سته های مورد استفاده بر روی دستگاه یا سرور فراتر از حداکثر آستانه مجاز نیست. دومین محدود کننده این اطمینان را به ما می دهد که هر دسته از پیکر بندی را می توان بر روی دستگاه یا سرور استفاده کرد. چارچوب به طور دینامیک می تواند با پیکر بندی سازگار شود و به این ترتیب از نظر هزینه

ارتباط کیفیت آم کاهش می یابد. با این حال، چارچوب نیازمند پیگر بندی های مختلف است. به علاوه، چارچوب به یک پشتیبانی رویکرد دروازه سرویس باز در ابر دور نیاز دارد.

ت: هیراکس: هیراکس (ماریندلی 2009) منابع موجود را از دسته ای از ابزار های موبایل در مجاورت محلی برای اجرای کار های محاسباتی اهرم بندی می کند. هیراکس از مکانیسم تحمل خطای هادوپ برای کاهش انفصالات از سرور های موبایل استفاده می کند. هیراکس امکان قابلیت دسترسی را به ابر های دورتر در منابع موبایل مربوط فراهم می کند و از این روی ممکن است به طور کامل قابل دسترس نباشد. هیراکس بر اساس هادوپ برای دستگاه های موبایل اندروید توسعه می یابد. سرور از دو فرایند طرف مشتری مپ ردیوس map reduce یعنی NameNode و JobTracker برای هماهنگ سازی محاسبه در گروهی از دستگاه های موبایل استفاده می شود. دستگاه موبایل از دو فرایند هادوپ استفاده می کند یعنی TaskTracker و DataNode برای دریافت وظایف از JobTracker. دستگاه های موبایل با سرور و دستگاه های موبایل دیگر ارتباط برقرار می کند. هیراکس به طور شفاف از منابع توزیعی استفاده کرده و یک قابلیت همکاری را در میان پلتفرم های ناهمگن ایجاد می کند. با این حال هیراکس دارای سربار بالایی به دلیل پیچیدگی الگوریتم هادوپ است.

ت: معماری افلود رپلیکاس موبایل مجازی (VMROA): در مقاله چان و مانیتیس (2009)، محققان یک معماری افلود را با استفاده از نسخه موبایل مجازی هم زمان سازی شده در ابر پیشنهاد کردند. معماری تقاضای پنهان باند کم تری به دلیل تکرار سیستم چک پوینت افزایشی دارد. سه مولفه اصلی از معماری وجود دارد یعنی تکرار گر، کنترل گر و تقویت گر. تکرار گر مسئول هم زمان سازی تغییرات در نرم افزار موبایل و وضعیت کلون است. کنترل گر بر روی دستگاه موبایل یک اجرای تقویتی را شروع کرده و نتایج را در دستگاه موبایل قرار می دهد. تقویت کننده در طرف کلون مسئول اجرای محلی و بازگشت نتایج است. VMROA سر بار هم زمان سازی را با هم زمان سازی فرصت طلبانه دستگاه موبایل و سرور ابری کاهش می دهد. با این حال، مهاجرت محیط اجرا موجب بروز مسائل امنیتی، حریم خصوصی و مدیریت در ابر می شود.

ت: MAUI: در مقاله چروو و همکاران (2010)، MAUI، که یک روش انرژی آگاه است، مکانیسم افلود در برنامه موبایل در ابر پیشنهاد می شود. MAUI از هر دو پارتیشن بندی دینامیک و استاتیگ استفاده می کند. اول، برنامه نویس روش را به صورت از راه دور تفسیر می کند که قادر به پیاده سازی و اجرای وظایف زیر نیست: 1- پیاده

سازی رابط کاربر و 2- پیاده سازی ورودی/خروجی/ از این روی سیستم پیشنهادی به طور خودکار روش های دور دست و وضعیت روش ها را شناسایی کرده و به طور خودکار مهاجرت را اجرا می کند. MAUI از مکانیسم تایم آوت برای تشخیص شکست در اتصال با سرور استفاده می کند. MAUI سه نوع پروفیل بندی را اجرا می کند یعنی پروفیل بندی دستگاه، پروفیل بندی برنامه و پروفیل بندی شبکه. MAUI مسئله اجرایی را به صورت مسئله برنامه نویسی خطی صحیح 0-1 فرموله می کند:

$$\sum_{v \in V} I_v \times E_v^l - \sum_{(u,v) \in E} |I_u - I_v| \times C_{u,v}$$

بیشینه سازی

$$\sum_{v \in V} ((1 - I_v) \times T_v^l) + (I_v \times T_v^r) + \sum_{(u,v) \in E} (|I_u - I_v| \times B_{u,v})$$

و به طوری که

$$\leq L \text{ and } I_v \leq r_v, \quad \forall v \in V \quad (10)$$

که I_v متغیر تصمیم گیری می باشد: در صورتی که روش به طور محلی اجرا شود، $I_v = 0$ می باشد در غیر این صورت، $I_v = 1$ می باشد. E_v^l و T_v^l نشان دهنده انرژی و زمان مورد نیاز برای اجرای محلی روش است. $B_{u,v}$ نشان دهنده وضعیت برنامه لازم می باشد در زمانی که u ، با v تماس بگیرد. پارامتر $C_{u,v}$ نشان دهنده هزینه انرژی حالت های انتقال است و پارامتر $C_{u,v}$ نشان می دهد که روش قابل دسترس است. اگرچه MAUI به طور معنی دار موجب بهبود مصرف دستگاه موبایل می شود و از تحرک کاربر و پویایی شبکه استفاده می کند که مقیاس پذیری را در نظر نگرفته و از این روی ویژگی های QoS را ندارد.

جدول 1 خلاصه ای مقایسه ای از چارچوب های مبتنی بر بهینه سازی تک منظوره را با در نظر گرفتن هدف نشان می دهد. مقایسه کلی این چارچوب ها در جدول 2 نشان داده شده است.

جدول 1: مقایسه چارچوب های مبتنی بر بهینه سازی تک منظوره بر اساس اهداف

کمینه سازی مصرف انرژی	ثبات منابع شفاف کاهش نهفتگی شبکه	بهینه سازی استفاده	بهینه سازی هزینه اجرا	بهینه سازی بازده	چارچوب های بهینه سازی تک منظوره
X	X	X	X	✓	CloneCloud (Chun et al., 2011)
X	X	X	X	✓	Data stream application partitioning framework (Yang et al., 2012)
X	X	✓	X	X	CODM (Verbelen et al., 2011)
X	✓	X	X	X	Hyrax (Marinelli, 2009)
X	✓	X	X	X	VMROA (Chun and Maniatis, 2009)
✓	X	X	X	X	MAUI (Cuervo et al., 2010)

جدول 2: مقایسه کلی چارچوب های مبتنی بر بهینه سازی تک منظوره

هزینه عملیاتی	مصرف انرژی	سربرابر ابر پشتیبان برنامه نویس	مقیاس پذیر ربر پروفیلر	QoS پشتیبانی	تاخیر شبکه	چارچوب های اجرای برنامه
High	Low	High	No	n/a	Low	CloneCloud (Chun et al., 2011)
Low	Low	Med.	Yes	No	No	Data stream application partitioning framework (Yang et al., 2012)
Low	Low	Low	No	Yes	Low	CODM (Verbelen et al., 2011)
Med.	Low	Low	Yes	n/a	Low	Hyrax (Marinelli, 2009)
High	Low	High	Yes	n/a	Low	VMROA (Chun and Maniatis, 2009)
Low	Low	Low	No	Yes	Low	MAUI (Cuervo et al., 2010)

4-2: چارچوب های مبتنی بر بهینه سازی دو منظوره

چارچوب های مبتنی بر بهینه سازی دو منظوره عمدتاً بر بهینه سازی دو هدف تاکید دارد. چارچوب های مبتنی بر بهینه سازی دو منظوره دارای پیچیدگی نسبتاً بیشتری نسبت به چارچوب های مبتنی بر بهینه سازی تک منظوره می باشند. این چارچوب ها از دو پارامتر در تابع هدف استفاده می کنند. چارچوب مبتنی بر بهینه سازی دو منظوره برازش و تناسب بهتری با محیط های متنوع دارد و می تواند ملزومات کاربرد در MCC را برآورده کند. از این روی ما چارچوب های بهینه سازی را برای بهبود دو تابع هدف ارایه می کنیم:

الف: تکه ابر مبتنی بر ماشین مجازی: در استایرینیان و همکاران (2009)، محققان به محدودیت های رایانش ابری اشاره کرده اند. نهفتگی WAN، جیتر، و از بین رفتن بسته برخی از این محدودیت ها هستند. استفاده از تکه ابر در مجاورت محلی موجب کاهش تاخیر شده و یک مرکز منفرد را ارایه می کند و دسترسی بی سیم پهنای باند به تکه ابر غنی از منبع را امکان پذیر می کند. دو روش برای تحویل VM به تکه ابر وجود دارد. در رویکرد مهاجرت VM، یک VM اجرایی متوقف می شود. همه حالت ها به تکه ابر انتقال می یابد و سپس اجرای VM

در تکه ابر شروع می شود. در روش ترکیب و سنتز VM، دستگاه موبایل یک VM کوچک را به تکه ابر ارسال می کند که قبلاً دارای VM پایه است که VM از آن گرفته شده است. راه حل پیشنهادی موجب کاهش سر بار مهاجرت با استفاده از ترکیب دینامیک می شود. به علاوه، عملکرد ترکیب با پردازش موازی با روش های مختلف بهبود می یابد. با این حال، کنترل دسترسی و حریم خصوصی مسائلی با مهاجرت محیط اجرایی کل است که باید به آن رسیدگی شود.

ب: AIOLOS: در مطالعه ورلن و همکاران (2012الف): AIOLOS یک چارچوب میان افزار است که دارای یک موتور تصمیمی است که منابع پویای سرور و شرایط شبکه متغیر را در تصمیم افلود در نظر می گیرد. چارچوب، زمان اجرا را برای هر روش در هر دو اجرای محلی و دور دست با در نظر گرفتن اندازه پوشش می دهد. ر اساس نتایج، موقعیت اجرا انتخاب می شود. هدف الگوریتم تصمیم افلود، بهینه سازی زمان اجرای محلی و مصرف انرژی است. پروفیل مبتنی بر تاریخچه برای روش سرویس به منظور محاسبه زمان اجرای محلی استفاده می شود که پردازنده الف، پهنای باند شبکه بتا و تاخیر گاما را در نظر می گیرد.

$$\bar{T}_{remote} = \frac{1}{\alpha} \times \sum_{i \in RM} (\bar{T}_{CPU,local_i}) + \frac{1}{\beta} \times (A+R) + \gamma + \sum_{j \in CM} \left(\bar{T}_{CPU,local_j} + \frac{1}{\beta} \times (A_j + R_j) + \gamma \right) \quad (11)$$

پارامترهای A-R نشان دهنده اندازه تقویت و اندازه نرخ بازده برآورد شده است:

$$\bar{E}_{saved} = E_{CPU} \times \sum_{i \in RM} (\bar{T}_{CPU,local_i}) - E_{TR} \times A - E_{RCV} \times R - \sum_{j \in CM} (E_{RCV} \times A_j + E_{TR} \times R_j) > 0 \quad (12)$$

که E_{CPU} ، E_{TR} و E_{RCV} نشان دهنده انرژی مصرف شده در هر واحد زمانی با سی پی یو، هزینه انرژی برای انتقال و دریافت یک بایت به ترتیب می باشد. AIOLOS تنها عملکرد بهتر در سناریو ها دارد که در آن تاریخچه قبلی سرور ابر قابل دسترس است.

پ: تماس ابر: یک چارچوب میان افزار در مقاله گیگوریا و همکاران (2009) ارائه شده است که به طور دینامیک یک برنامه را پارتیشن بندی کرده و پارتیشن ها را به ابر افلود می کند. فرایند پارتیشن بندی شامل دو مرحله

است. اول، یک برنامه به صورت یک گراف جریان داده مدل سازی می شود که متشکل از ماژول های نرم افزار مختلف می باشد. از این روی، یک الگوریتم پارتیشن بندی، کاهش بهینه را محاسبه می کند که قادر به بهینه سازی تابع هدف است. تابع هدف بهینه سازی، تاخیر تعامل را بین دستگاه موبایل و سرور ابر به حداقل می رساند ضمن این که سربار داده های مبادله شده را نیز در نظر می گیرد:

$$\min O_{cc} = \min \left(\sum_{i=1}^t \sum_{j=1}^w \frac{in_{ij} + out_{ji} * f_{ij}}{\alpha} + \sum_{i=1}^k \frac{code_size_i}{\beta} + \sum_{i=1}^{w < s} proxy_cost_i \right) \quad (13)$$

که اولین بخش معادله 3، هزینه تبادل داده ها را بین اجزای تخصیص داده نشده مدل سازی می کند. پارامتر f_{ij} نشان دهنده فرکانس تبادل داده بین اجزای تخصیص داده نشده است. الفا و بتا بیانگر ظرفیت ها و منابع محاسباتی می باشد. دومین جمله نشان دهنده هزینه محاسباتی بر روی دستگاه موبایل است آخرین جمله نشان دهنده هزینه ایجاد پروکسی هایی برای تعامل با دسته های دور W است. این چارچوب مصرف حافظه، هزینه ارتباطی و زمان تعامل را کاهش می دهد. با این حال، این چارچوب از فرایند مهاجرت رایانشی استفاده می کند که در تحلیل دینامیک، پروفیل بندی، سنتز، پارتیشن بندی و افلودینگ نقش دارد. این چارچوب نیازمند هم زمان سازی پیوسته است که دستگاه موبایل را برای کل پلتفرم توزیعی در حالت فعال قرار می دهد.

ت: چارچوب اجرای ابر موبایل: در مقاله هوانک و همکاران (2012) یک چارچوب اجرای برنامه مبتنی بر ابر نیازمند باز طراحی برنامه نیست.

بر خلاف مهاجرت برنامه طرح مبتنی بر VM، طرح پیشنهادی تنها مستلزم انتقال حالت برنامه ذخیره شده به جای کل حالات VM است. برای مهاجرت یک برنامه، چارچوب ابتدا برنامه را روی موبایل متوقف می کند و فایل داده های حالت ذخیره شده را با استفاده از آن در ابر ارسال کرده و در نهایت اجرای برنامه را در سرور ابری به طور مجدد شروع می کند. وقتی که فایل های داده های حالت مند دارای اندازه کوچک تری هستند، سربار کوچک تر می شود. برای اجتناب از از بین رفتن داده های ورودی، روش پاسخ برنامه با طرح ذخیره حالت ترکیب می شود. چارچوب پیشنهادی یک کد امنیتی را از طریق رمز گذاری ایجاد کرده و محیط اجرایی را ایزوله می کند. به علاوه، عملکرد برنامه های تعاملی در شبکه های با پهنای باند پایین کاهش می یابد.

پ: محیط خدمات پایانه های مشارکتی به کمک تکه ابر: CACTSE (گینگ و همکاران 2013) به صورت یک چارچوب تحویل محتوی مبتنی بر تکه ابر پیشنهاد می شود. این چارچوب دارای مدیریت سرویسی است که توزیع محتوی را میان گره های موبایل مدیریت می کند. تکه ابر داده های واقعی را ذخیره نمی کند بلکه محتوی را شاخص بندی می کند. دستگاه های موبایل خود را با مدیر سرویس ثبت کرده و درخواست داده هایی از سرور می کند. مدیر سرویس ابتدا شاخص محلی را برای فایل درخواست شده جست و جو می کند. دستگاه های موبایل خود را با مدیر سرویس ثبت کرده و سپس داده ها را از سرور درخواست می کنند. در صورت یافت شاخص، مدیر سرویس گره درخواستی را به گره دیگر هدایت می کند که حاوی فایل درخواستی است. اگر داده های درخواستی بر روی حافظه گره محلی موجود نباشد، مدیر سرویس به اینترنت وصل شده و فایل داده های مورد نیاز را دانلود می کند. هدف CACTSE، ایجاد هماهنگی میان گره های موبایل محلی برای مبادله داده ها بدون اینترنت یا سرویس ابر است. CACTSE محتویات را به کاربران بدون اختلال در شبکه های تحلیل محتوی هل می دهد. چارچوب، تقاضای آینده را بر اساس رفتار کاربر پیش بینی می کند و از این روی محتوی به حاشیه در حالت منفعل ارسال می شود. با این حال CACTSE موجب افزایش بار مدیریت در شبکه دسترسی می شود. محتوی ذخیره شده با گذشت زمان قدیمی تر می شود

ج: دیکشنری چند زبانه مبتنی بر تکه ابر: یک دیکشنری چند زبانه مبتنی بر ابر در مقاله اکانتا و همکاران (2012) پیشنهاد شده است. دیکشنری چند زبانه فوق، زیر ساخت تکه ابر مبتنی بر VM را برای اجرای برنامه اهرم بندی می کند. هدف راه حل پیشنهادی کمینه سازی مهاجرت سر بار VM با استفاده از ترکیب VM می باشد. ترکیب VM شامل دو VM موسوم به VM پایه ای دیکشنری و VM فوقانی است. VM پایه درون تکه ابر قرار داشته و نوع دوم درون دستگاه موبایل است. VM فوقانی دیکشنری از دستگاه موبایل به سرور ابر مهاجرت می کند که در آن دیکشنری رویی به نوع زیرین منتقل می شود.

$$\min O_{C_c} = \min \left(\sum_{i=1}^{t < k} \sum_{j=1}^w \frac{\text{in}_{ij} + \text{out}_{ji} * f_{ij}}{\alpha} + \sum_{i=1}^k \frac{\text{code_size}_i}{\beta} + \sum_{i=1}^{w < s} \text{proxy_cost}_i \right) \quad (13)$$

که اولین بخش معادله (13)، هزینه مبادله داده ها بین مولفه ها و اجزای غیر نزدیک مدل سازی می کند. پارامتر f_{ij} نشان دهنده فراوانی مبادله داده ها بین اجزای غیر عددی است. آلفا و بتا نشان دهنده قابلیت های منابع

ارتباطی و محاسباتی است. دومین عبارت نشان دهنده هزینه محاسباتی بر روی دستگاه موبایل است. آخرین مورد نشان دهنده هزینه ایجاد پروکسی ها برای تعامل با دسته های W دور است. با این حال چارچوب فوق از فرایند مهاجرتی نیازمند محاسبه و رایانش استفاده می کند.

ح: معکوس سازی دستگاه موبایل: زائو و همکاران چارچوب مبتنی بر سرور آینه را پیشنهاد کرده اند که یک VM را برای دستگاه های موبایل درون شبکه عرضه کننده سرویس ارتباط از راه دور ارایه می کند. چارچوب از بار کاری به آینه دستگاه موبایل در سرور ابری تغییر می کند. این سرور از هم زمان سازی بین دستگاه موبایل و سرور آینه پشتیبانی کرده و از این روی به ورودی های دستگاه موبایل به شکل معکوس پاسخ می دهد. پشتیبانی از ذخیره داده ها بر روی آینه امکان حفظ فایل های دانلود شده از اینترنت به حافظه آینه را می دهد. فایل ها را می توان به طور مستقیم از کاربران بعدی از حافظه سرور از سرور اینترنت بازیابی می شود. به طور مشابه وقتی که کاربر یک فایل را به کاربران مختلف ارسال می کند، دستگاه موبایل نیازی به ارسال فایل ها به لینک بی سیم ندارد. دستگاه های موبایل تنها فایلی را به سرور آینه ارسال کرده و سرور آینه دستگاه های موبایل را بر اساس درخواست ارسال می کند. آینه بندی دستگاه موبایل موجب کاهش سر بار عملیاتی یک دستگاه موبایل با انتقال عملکرد آپلود به سرور آینه می شود. این چارچوب سر بار هم زمان سازی بالایی را ایجاد می کند که نیازمند پهنای باند شبکه ای بیشتری است.

چ: آلود کد با اجرای مهاجرت شفاف (COMET): کومت بر مهاجرت شفاف برنامه چند رشته ای به سرور های قابل دسترس محلی تاکید می کند. این چارچوب از تصمیم مهاجرت با در نظر گرفتن بار کار ماشین استفاده می کند. کومت از روش های حافظه مشترک توزیعی نظیر گرانولیته سطح میدانی، برای حفظ پیوستگی میان سیستم های نهایی استفاده می کند. رشته های کاربرد مخاطبان و نویسندگان در کومت به طور هم زمان از فیلد بدون هماهنگی استفاده می کند. کومت یک هم زمان سازی VM را بین دستگاه موبایل و سرور ابر ارایه می کند. یک برنامه نویس در کومت، رشته ها را بین نقاط انتهایی برای بهینه سازی بازده مهاجرت می دهد. این برنامه نویس، رفتار گذشته اجرای رشته را در تصمیم مهاجرت رشته در نظر می گیرد. رفتار گذشته با پایش میزان اجرای رشته بر روی دستگاه موبایل بدون استفاده از روش های بومی در نظر می گیرد. کومت مهاجرت شفاف

رشته های جزئی و چند رشته های کاربرد را در نظر می گیرد و از این روی این چارچوب مکانیسم امنیتی ارایه نکرده و از یکپارچگی داده ها اطمینان حاصل نمی کند.

ر: چارچوب برنامه تکراری: در لی (2012)، چارچوب مبتنی بر تکرار برنامه پیشنهاد می شود که شامل گره رئیس و گره کارگر است. گره های کارگر بر روی کلاس های مهاجرت شده اجرا می شوند در حالی که گره های رئیس، درخواست های افلود را از مشتریان موبایل دریافت کرده و آن ها را به گره های کارگر مناسب ارسال می کند. برای کاهش تاخیر، درخواست اول دارای یک شناساگر کلاس می باشد با این حال وقتی گره پاسخ سرور ابر در معرض نبود نسخه در ابر قرار می گیرد، فایل های دو دویی از مشتری موبایل به گره رئیس مهاجرت می کند. این چارچوب سربار مهاجرت را با تکرار برنامه درون ابر برای استفاده بعدی کاهش می یابد. تصمیم افلود سبک وزن است زیرا تصمیم گیری درون ابر به جای دستگاه موبایل انجام می شود. با این حال تصمیم افلود در ابر نیازمند اطلاعاتی از دستگاه موبایل نظیر ارتباط شبکه، سلیق کاربر و قابلیت های دستگاه موبایل است.

خ: کوکو: کمپ و همکاران 2012 یک چارچوبی را برای افلود جزئی برنامه های موبایل به سرور های ابر پیشنهاد می کند. هدف کوکو، تسهیل توسعه دهنده ها با اهرم بندی ابزار های توسعه می باشد که برای توسعه دهنده ها مشهود است.

کوکو متشکل از دو نوع سازنده است: دوباره نویسی سرویس کوکو و درایور سرویس از راه دور کوکو. کوکو اجزای نیازمند محاسبه و تعامل برنامه را با استفاده از زبان تعریف رابط اندروید تفکیک می کند. وقتی که سرویس بر روی سرور موجود است، ادرس سرور به مدیر منبع اجرا شده بر روی دستگاه موبایل ارسال می شود. سپس، رجیستر، ادرس سرور را ثبت می کند تا دستگاه موبایل را قادر به استفاده از منابع کند. چارچوب کوکو نیازمند پشتیبانی برنامه نویسی برای اصلاح برنامه است. کوکو تصمیمات ساکن را برای افلود می گیرد. به علاوه، کوکو اجرای برنامه مبتنی بر پروکسی را ارایه می کند که بر تاخیر افزایشی تاکید دارد.

جدول 3: مقایسه چارچوب های مبتنی بر بهینه سازی دو منظوره بر اساس اهداف

	کمینه سازی زمان پاسخ سریار مهاجرت	کمینه سازی بهینه سازی مصرف باند	بهینه سازی هزینه انتقال داده	کاهش مصرف انرژی	بیشینه سازی مصرف منابع	کاهش تاخیر شبکه	کاهش زمان افلود	کمیته سازی زمان اجرا	بهینه سازی هزینه اجرا	چارچوب مبتنی بر بهینه سازی دو منظوره
VM-based cloudlet (Satyanarayanan et al., 2009)	X	✓	X	✓	X	X	X	X	X	X
AIOLOS (Verbelen et al., 2012a)	X	X	X	✓	X	✓	X	X	X	X
Calling the cloud (Giurgiu et al., 2009)	X	X	X	✓	X	X	✓	X	X	X
Mobile cloud execution framework (Hung et al., 2012)	X	X	✓	X	X	X	✓	X	X	X
CACTSE (Qing et al., 2013)	X	X	X	✓	X	X	✓	X	X	X
Cloudlet-based multi- lingual dictionaries (Achanta et al., 2012)	X	X	X	✓	X	X	X	X	✓	X
Dynamic cloudlets (Verbelen et al., 2012b)	X	X	X	✓	✓	X	✓	X	X	X
Virtual mobile cloud computing (VMCC) (Huerta-Canepa and Lee, 2010)	X	X	X	✓	✓	X	✓	X	X	X
Mirroring mobile device (Zhao et al., 2012)	✓	X	X	✓	X	X	X	X	X	X
COMET (Gordon et al., 2012)	X	✓	X	X	X	✓	X	X	X	X
Replicated application framework (Lee, 2012)	X	X	X	✓	X	✓	X	X	X	X
Cuckoo (Kemp et al., 2012)	X	✓	X	X	X	✓	X	X	X	X
MOCHA (Soyata et al., 2012)	X	X	X	✓	X	X	X	X	X	✓

جدول 4: مقایسه کلی چارچوب های مبتنی بر بهینه سازی دو منظوره

چارچوب اجرای برنامه	تخریبی	قابلیت پشتیبانی	دسترسی بروزرسانی	تعمیر پذیری	امنیت داده	سرور ایر	صرف انرژی	تعمیر پذیری
VM-based cloudlet (Satyanarayanan et al., 2009)	Low	n/a	n/a	Yes	n/a	Med.	Low	Med.
AIOLOS (Verbelen et al., 2012a)	Low	No	Low	No	No	Low	Low	Low
Calling the cloud (Giurgiu et al., 2009)	Low	n/a	High	No	No	Low	Low	Low
Mobile cloud execution framework (Hung et al., 2012)	High	Yes	Low	No	Yes	Low	Low	Low
CACTSE (Qing et al., 2013)	Low	Yes	n/a	Yes	No	Low	Low	Low
Cloudlet-based multi-lingual dictionaries (Achanta et al., 2012)	Low	n/a	n/a	Yes	n/a	Med.	Low	Low
Dynamic cloudlets (Verbelen et al., 2012b)	Low	Yes	High	Yes	Yes	Low	Low	Low
Virtual mobile cloud computing (VMCC) (Huerta-Canepa and Lee, 2010)	Low	No	n/a	No	No	High	Low	Low
Mirroring mobile device (Zhao et al., 2012)	Low	n/a	n/a	Yes	n/a	High	Low	High
COMET (Gordon et al., 2012)	Low	n/a	n/a	Yes	Less	Low	Low	Low
Replicated application framework (Lee, 2012)	Low	No	Yes	Yes	n/a	High	Low	High
Cuckoo (Kemp et al., 2012)	Low	n/a	n/a	Yes	n/a	Low	Low	Low
MOCHA (Soyata et al., 2012)	High	Yes	n/a	Yes	n/a	Med.	Med.	High

ص: معماری شتاب ابر- تکه ابر- موبایل(MOCHA):MOCHA در مقاله سویوتا و همکاران(2012) برای اهرم بندی بر روی منابع تکه ابر تقریبی موبایل پیشنهاد میشود. معماری MOACHA متشکل از سه مولفه است: دستگاه موبایل، تکه ابر و ابر. دستگاه موبایل به تکه ابر متصل است که تکه ابر به یک زیر ساختار بزرگ تر نظیر سرویس وب آمازون، ویندوز آزور متصل است. تکه ابر شیوه پارتیشن بندی محاسبه را میان سرور ها در ابر بر اساس شخص های QoS مختلف پیدا می کند. MOACHA موجب کاهش تاخیر از کاربر به ابر می شود. MOCHA از پردازش موازی برای اجرای برنامه پشتیبانی کرده و QoS را ارایه می کند. با این حال دست یابی به پارامتر شبکه QoS و یافتن بهترین مسیر آن، انرژی دستگاه موبایل را مصرف می کند.

جدول 3 خلاصه ای از چارچوب های مبتنی بر بهینه سازی دو منظوره را با در نظر گرفتن هدف ارایه می کند. مقایسه کلی این چارچوب ها در جدول 4 نشان داده شده است.

4-3 چارچوب های مبتنی بر بهینه سازی چند منظوره

چارچوب های مبتنی بر بهینه سازی چند منظوره عمدتاً بر بهینه سازی اهداف مختلف تاکید دارد. چارچوب های مبتنی بر بهینه سازی چند منظوره دارای بیشترین پیچیدگی نسبت به مقوله های دیگر، چارچوب های مبتنی بر بهینه سازی تک منظوره و چارچوب های مبتنی بر بهینه سازی دو منظوره می باشد. این چارچوب ها از طیف وسیعی از پارامتر ها در تابع هدف استفاده می کنند. چارچوب های مبتنی بر بهینه سازی چند منظوره بهتر از سایرین در محیط اجرای متنوع و رفع نیاز های برنامه های مختلف در MCC می باشند. از این روی ما چارچوب هایی را ارایه می دهیم که توابع هدف را بهینه سازی می کنند.

1- مدل برنامه انعطاف پذیر: در مقاله زانگ(2010)، یک مدل برنامه الاستیک ارایه می شود که امکان استفاده از منابع ابری را به طور شفاف می دهد. چارچوب انعطاف پذیر پیشنهادی از چهار صفت در مدل هزینه استفاده می کند: کمینه سازی مصرف برق، بیشینه سازی هزینه پولی، بیشینه سازی سود و بیشینه سازی امنیت و حریم خصوصی. این برنامه به اجزای مختلف موسوم به ولت تقسیم می شود که در ابرهای مختلف تکرار می شود به علاوه پیکر بندی اجرای برنامه با تابع هدف زیر بهینه سازی می شود:

$$y^* = \arg \max_y p(y) \prod_{i=1}^L p(x_i|y) \prod_{j=1}^M p(z_j|y) \quad (14)$$

بردار 'x' دارای مقادیر با اجزای وضعیت دستگاه متفاوت نظیر بازده، مصرف حافظه، پهنای باند اپلود و ذخیره فایل می باشد. بردار Z دارای مقادیری برای گزینه های مطلوب ابر نظیر سرعت پردازش و هزینه پولی است. متغیر Y یک متغیر پیکر بندی می باشد که نشان دهنده تعداد کل پیکر بندی ها است. متغیر های L-M نشان دهنده تعداد اجزا در بردار وضعیت و تعداد اجزا در بردار اولویت به ترتیب می باشد. ایجاد زمان اجرای بستر توزیعی و مدیریت نیازمند منابع محاسباتی اضافی است. تصمیم افلود بر اساس مدل هزینه پیچیده است که از پارامتر ها بهره می برد و از این روی تصمیم گیری هایی را انجام می دهد.

2- تینک ایر(ThinkAir) از مجازی سازی دستگاه متبایل برای اجرای هم زمان روش های افلود مختلف ریاضی استفاده کرده و از این روی زمان اجرای برنامه را کاهش می دهد. درخواست روش از راه دور از طریق کنترل گر اجرا صورت می گیرد و بر روی دستگاه موبایل و ابر قرار می گیرد. کنترل گر اجرا، تصمیم را بر اساس اطلاعات جمع آوری شده توسط نگارنده پروفیل می گیرد. در صورتی که ارتباط از راه دور با مشکل مواجه شود، چارچوب قادر به اجرای محلی نخواهد بود. سمت ابری کد افلود شده توسط سرور برنامه مدیریت می شود. چارچوب از شش نوع VM با CPU متفاوت، و حافظه پشتیبانی می کند. سرور پیش فرض موسوم به سرور اولیه است و از این روی معمولاً آنلاین است و بقیه شامل نوع سرور ثانویه نیز هست. به علاوه چارچوب از پروفیل بندی صحیح و سبک وزن برای سخت افزار، نرم افزار و شبکه پشتیبانی می کند. چارچوب از اجرای موازی پشتیبانی کرده و سناریو های افلود چند کاربری را برای اجرا در ابر نظر گرفته می شود و از این روی چارچوب نیازمند نصب VM مبتنی بر سیستم عامل اندروید در سرور ابر می باشد.

3- تکه ابر بسته: یک تکه ابر مبتنی بر حافظه در مقاله کودولیت و همکاران(2012) ارائه شده است که هدف آن استفاده از ظرفیت ذخیره ای موجود دستگاه های موبایل برای کاهش تاخیر و مسائل انرژی در دست یابی به سرویس های ابری از راه دور است. چارچوب بر روی هر دو مدل دسترسی جامعه و مدل دسترسی شخصی برای بیشینه سازی تعداد پرس و جوی کاربر اهرم بندی می شود. سرویس های ابری کامل یا جزئی به دستگاه موبایل مهاجرت می کنند از این روی دستگاه موبایل به یک تکه ابر بسته تبدیل می شود. تکه ابر موجب بهبود تجربه کاربر با اجرای وظایف زیر می شود: 1- ذخیره اطلاعات در دستگاه موبایل 2- شخصی سازی سرویس بر اساس رفتار و الگو های مصرف کاربران و 3- اطمینان از حریم خصوصی کاربران با اجرای محلی خدمات بر روی دستگاه

موبایل. تکه ابر موجب بهبود تجربه کاربر با ارایه دسترسی فوری به اطلاعات می شود. ابر بسته موجب کاهش زمان کشف خدمات و مصرف انرژی می شود. با این حال، تکه ابر بسته نیازمند آپدیت های زمان واقعی در لینک های رادیویی برای اطمینان از تازگی داده های ذخیره شده است.

4- MISCO MISCO (دو و همکاران 2010)، MapReduce را به محیط ابر توزیعی توسعه داده است که متشکل از گره های کارگر موبایل و سرور مرکزی است. MapReduce چارچوب پردازش داده ها است که امکان اجرای موازی برنامه در محیط خوشه می شود. یک چارچوب MapReduce بر روی سرور اصلی مرکزی پیاده سازی می شود. ویژگی منحصر به فرد چارچوب این است که دستگاه های موبایل گره های کارگر می باشند و نتایج را به سرور اصلی باز می گردانند. میسکو از مکانیسم پایش مرکزی برای پایش پلتفرم اجرای توزیعی استفاده می کند. میسکو یک چکیده برنامه نویسی قوی را فراهم کرده و از پردازش اطلاعات موازی پشتیبانی می کند. با این حال، دستگاه های موبایل دارای سربار انتقال بالا برای هم زمان سازی و پایش ورودی و مبادله نتایج می باشند.

5- میان افزار ابر موبایل مبتنی بر پروتوکل حضور و پیام رسانی قابل توسعه : یک میان افزار ابر موبایل مبتنی بر XMPP در مقاله کاموچ و همکاران (2012 الف) ارایه شده است که برنامه پارتیشن بندی و افلود تطبیقی را به سرور های موجود داده است. پوشش میان افزار پیشنهادی دو بعد معماری رایانش ابری موبایل را پوشش می دهد: یک مدل هزینه زمینه آگاه و استاندارد سازی XMPP به عنوان پروتکل تکه ابر. تصمیم افلود بر اساس مدل هزینه زمینه آگاه است که شامل پارامتر های مختلفی است که زمان اجرا، سطح باطری، مصرف منبع، امنیت، هزینه پولی و پهنای باند شبکه را تنظیم می کند. هدف مدل اجرای تصمیمات هوشمند با سربار حداقل است ضمن این که محدودیت های محیط را کاهش می دهد. مدل بهینه سازی در XMPP با الگوریتم های تصمیم پیش بین، کارآمد و سبک پشتیبانی می شود. XMPP موجب کاهش تاخیر WAN، تاخیر انتقال شده، از QoS استفاده کرده و از این روی از سربار پروفیل بندی رنج می برند. افلود همانند الگوریتم پیش بینی سبک وزن است که بر روی دستگاه موبایل اجرا می شود در حالی که الگوریتم بهینه سازی در تکه ابر اجرا می شود. با این حال، میان افزار نیازمند پشتیبانی برنامه نویس برای تفسیر روش ها برای اجرا در MCC است.

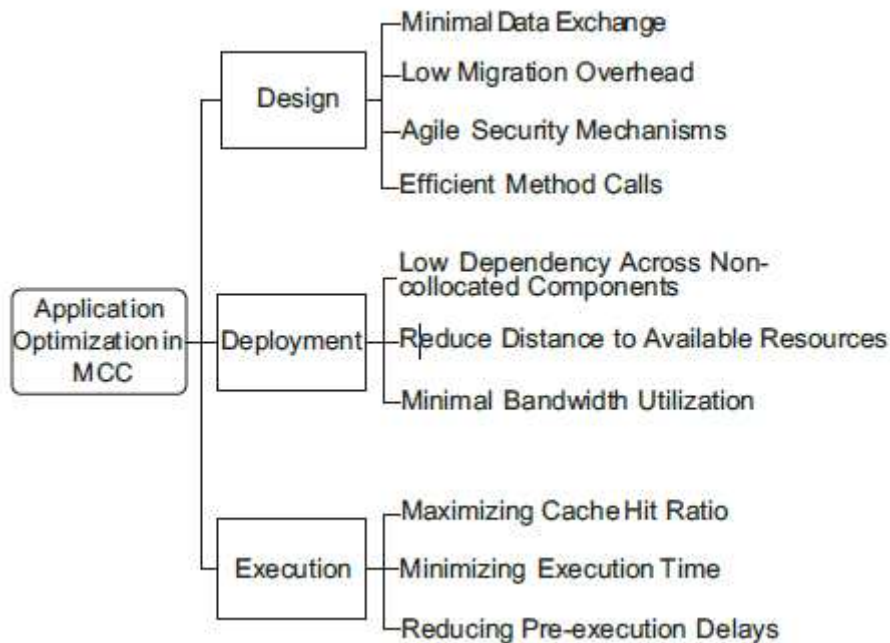
6- MACS: یک میان افزار تطبیقی که پارتیشن بندی برنامه سبک وزن، افلود محاسباتی و پایش منابع را در اختیار می گذارد در منبع کوچ و همکاران (2012 ب) ارایه شده است. تصمیم پارتیشن بندی به مسئله بهینه سازی

تبدیل شده و توسط حل کننده بهینه سازی حل می شود. تابع هزینه در مسئله بهینه سازی شامل هزینه انتقال خدمات، خدمات مرتبط با هزینه است که شامل هزینه CPU، و هزینه حافظه دستگاه است. محدودیت ها به کمینه سازی مصرف حافظه، مصرف انرژی، و زمان اجرا مربوط هستند. تابع هزینه به صورت زیر است

$$\min_{x \in 0,1} (C_{transfer} * W_{tr} + C_{memory} * W_{mem} + C_{CPU} * W_{CPU}) \quad (15)$$

جدول 6: مقایسه عمومی چارچوب های مبتنی بر بهینه سازی چند منظوره

چارچوب اجرای برنامه	تأثیر شبکه	قابلیت پشتیبانی	دسترسی بر روی فایبر	تعمیرات فایبر	پشتیبان برنامه نویسی	سربرابر	مصرف انرژی	هزینه عملیاتی
Elastic application model (Zhang et al., 2010)	High	Yes	High	Yes	Yes	High	Low	High
ThinkAir (Kosta et al., 2012)	n/a	No	High	Yes	Yes	High	Low	Low
Pocket Cloudlet (Koukoumidis et al., 2012)	Low	No	Med.	Yes	n/a	Low	Low	Low
MISCO (Dou et al., 2010)	Low	No	High	Yes	n/a	Low	High	Low
XMPP-based mobile cloud middleware (Kovachev et al., 2012a)	Med.	Yes	High	Yes	Yes	Low	Low	Med.
MACS (Kovachev et al., 2012b)	Low	No	High	No	Yes	Low	Low	Low



شکل 2: رده بندی راهبرد های بهینه سازی برنامه در MCC

که

$$C_{transfer} = \sum_{i=1}^n code_i * x_i + \sum_{i=1}^n \sum_{j=1}^k tr_j * (x_j \text{ XOR } x_i) \quad (16)$$

$$C_{memory} = \sum_{i=1}^n mem_i * (1 - x_i) \quad (17)$$

$$C_{CPU} = \sum_{i=1}^n code_i * \alpha * (1 - x_i) \quad (18)$$

در معادلات فوق، n تعداد ماژول های افلود شده است. برای یک ماژول خاص i ، اندازه کد و هزینه حافظه با $code_i$ و mem_i نشان داده می شوند. اندازه انتقال ماژول i با tr_i نشان داده می شود که برابر با جمع $send_i$ و rec_i است. x_i متغیر تصمیم گیری است که نشان می دهد آیا ماژول i به طور محلی اجرا می شود ($x_i = 0$) و یا به طور غیر محلی ($x_i = 1$). اگرچه MACS سبک وزن بوده و پارتیشن بندی دینامیک ارایه می کند، با این حال نیازمند پشتیبانی توسعه دهنده برای ساختار دهی در یک مدل است. به علاوه فرایند افلود با پروفیل بندی، پارتیشن بندی و سربرار مهاجرت تحت تاثیر قرار می گیرد.

جدول 5 خلاصه ای تفصیلی از چارچوب های مبتنی بر بهینه سازی چند منظوره را با در نظر گرفتن هدف ارایه می کند. مقایسه کلی این چارچوب ها در جدول 6 نشان داده شده است.

5- رده بندی برنامه محور برنامه موبایل مبتنی بر ابر

در این بخش، ما سه رده بندی مربوط به برنامه موبایل مبتنی بر ابر در MCC ارایه می کنیم. ابتدا رده بندی راهبرد های بهینه سازی را طبقه بندی می کند که برای بهینه سازی عملکرد برنامه استفاده می شود. دوم، رده بندی، عملیات دخیل را در طی اجرای برنامه در MCC طبقه بندی می کند.

5-1 رده بندی راهبرد های بهینه سازی برنامه در MCC

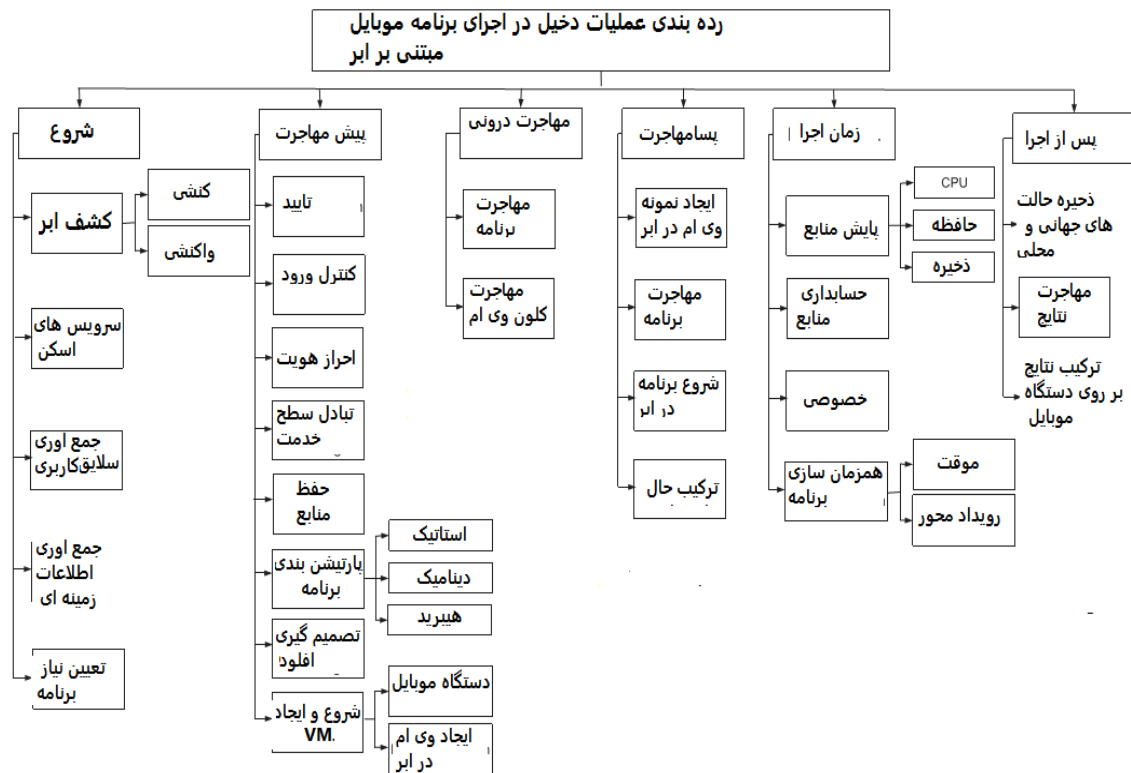
برنامه در MCC را می توان به سه بعد بهینه سازی کرد. 1- طراحی 2- بهره برداری 3- اجرا. شکل 2 رده بندی راهبرد های بهینه سازی مرتبط با برنامه را در MCC نشان می دهد. طراحی برنامه را می توان با استفاده از ویژگی های زیر بهینه سازی کرد. 1- کاهش تبادل داده ها 2- استفاده از چارچوب های مهاجرت سبک وزن 3- اهرم بندی مکانیسم های امنیت ناب و 4- کاهش سربرار روش. تبادل داده ها را می توان با کاهش وابستگی اجزا در طراحی برنامه کمینه سازی کرد. چارچوب های سبک وزن را می توان با کاهش سربرار زمان اجرا یا تغییر کارکرد

های غیر تعاملی از دستگاه موبایل به طرف ابر طراحی کرد. مکانیسم امنیت چابک یا ناب را می توان در طراحی برنامه با اتکا به امنیت ابر نظیر اعتماد مبتنی بر اعتبار یا استفاده از رمزگشا های سر بار پایین استفاده کرد. سر بار را می توان با استفاده از دستور مخاطبین یا با کاهش تعداد تماس های درخواستی با استفاده از عملیات مشابه در یک روش کاهش داد.

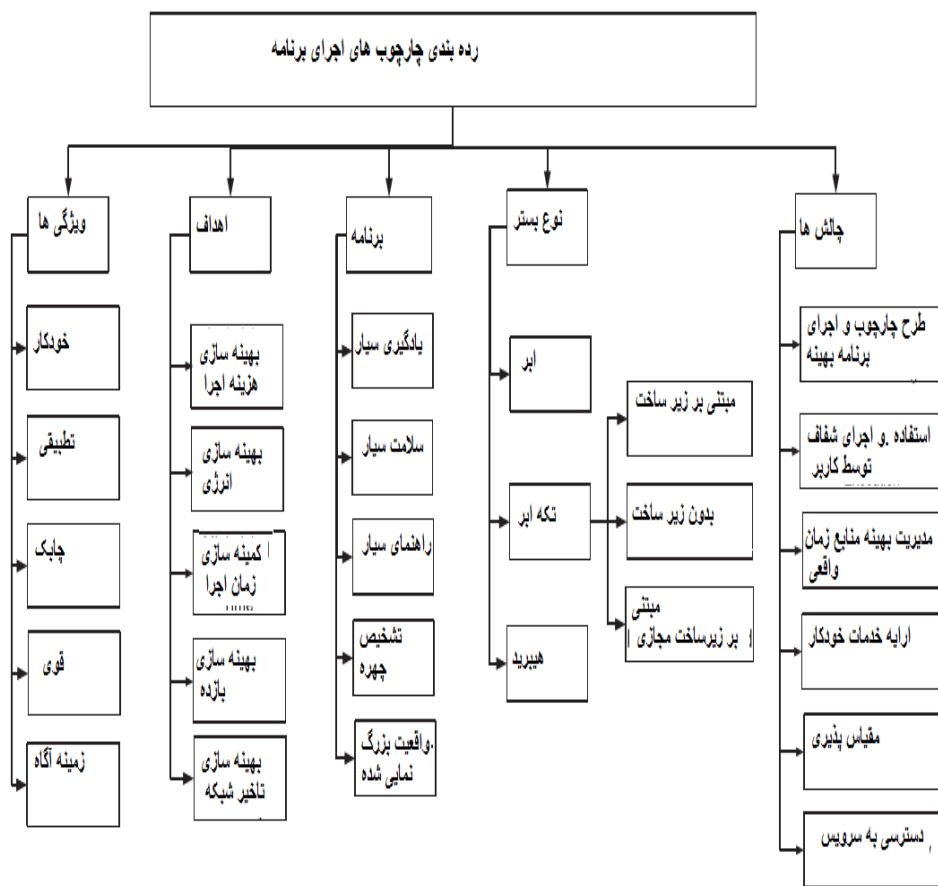
استفاده از برنامه را می توان با استفاده موثر از اجزا در محیط اجرایی بهینه سازی کرد. استفاده از برنامه موثر را می توان با در نظر گرفتن مسائل زیر به کار برد: 1- وابستگی میان اجزای غیر همسان باید کمینه سازی شود 2- فاصله تا منابع قابل دسترس 3- استفاده از پنهان باند باید حداقل باشد 4- هزینه زمان اجرا باید پایین تر باشد. اجزای زیر باید بر روی دستگاه موبایل اجرا شود. 1- رابط کاربر 2- ولفه دارای وابستگی سخت افزار بر روی دستگاه موبایل 3- مولفه های زمان واقعی و 4- اجزایی که نیازمند داده های امن هستند. اجرای برنامه در MCC را می توان با استفاده از ویژگی های زیر در محیط اجرایی بهینه سازی کرد. 1- ذخیره 2- اجرای موازی 3- پیش نصب. حافظه می تواند به کاهش زمان پاسخ برنامه با استفاده از داده ها و ذخیره برای استفاده های بعدی کمک کند. اجرای موازی موجب کاهش زمان اجرای کلی در ابر می شود ولی نیازمند زمان بندی موثر است. پیش نصب موجب کاهش تاخیر پیش اجرا در MCC می شود. شکل 3 رده بندی عملیات اجرای برنامه را در MCC نشان می دهد.

5-2 رده بندی عملیات اجرای برنامه در MCC

این بخش به بررسی رده بندی عملیات اجرا شده با چارچوب های اجرای برنامه در MCC می پردازد. شکل 3 رده بندی عملیات اجرای برنامه را در MCC نشان می دهد. عملیات دخیل در اجرای برنامه های مبتنی بر ابر به 5 گروه طبقه بندی می شوند: آغاز، پیش مهاجرت، در هنگام مهاجرت، پس از مهاجرت، در حین اجرا و پس از اجرا.



شکل 3: رده بندی عملیات اجرای برنامه در MCC



شکل 4: رده بندی چارچوب های اجرای برنامه

عملیات شروع مربوط به کشف ابر، جمع آوری سلائیق کاربر، جمع آوری اطلاعات زمینه ای و بررسی نیاز های برنامه است. کشف ابر فرایند یافتن سرور ابر برای اهرم بندی منابع موجود بر روی سرور برای اجرای برنامه می باشد. کشف ابر را می توان به دو روش به شکل کنشی و واکنشی انجام داد. کشف ابر کنشی موجب کاهش تاخیر پیش از اجرا می شود، در حالی که کشف ابر واکنشی منجر به تاخیر قبل از شروع ارائه خدمات توسط ابر می شود. اسکن خدمات و سرویس با درخواست کاربر انجام می شود. از این روی عملیات اسکن سرویس یک کار زمان بر است. عملیات جمع آوری سلائیق کاربر، سلائیق کار بر در خصوص هزینه، کیفیت و شاخص های عملکردی دیگر جمع آوری می کند. اطلاعات زمینه ای مربوط به شبکه و دستکاه در عملیات شروع جمع آوری می شود. اطلاعات زمینه ای شامل بار کار فعلی و توان باطری باقی مانده است نیاز های برنامه در طی شروع برنامه جمع آوری و مشخص می شود.

پیش مهاجرت شامل احراز هویت، کنترل ورود و تبادل توافق سطح خدماتی، حفظ منابع، پارتیشن بندی منابع، تصمیم افلود، ایجاد نمونه VM و شروع آن است. احراز هویت نشان می دهد که یک فرد همان چیزی است که ادعا می کند. کنترل ورود فرایند ارزیابی است که کنترل می کند آیا منابع ابری برای اجرای برنامه درخواستی قبل از شروع اجرای واقعی کافی است یا نه. احراز هویت فرایند تعیین حقوق دسترسی به منابع ابری است. توافق سطح خدماتی بین کاربران موبایل و عرضه کننده خدمات ابر برای مبادله نیاز های منابع و سیاست های خدماتی صورت می گیرد. فرایند ذخیره منابع، منابع سرور ابر را برای کاربر موبایل بر اساس توافق نامه ذخیره می کند. پارتیشن بندی منابع فرایندی است که شامل تقسیم یک برنامه به اجزای قابل انتقال و غیر قابل انتقال است. پارتیشن بندی منابع در حالت ایستا، پویا و ترکیبی انجام می شود. دیگر عملیات مهم در فاز قبل از مهاجرت تصمیم گیری است. تصمیم گیری افلود با در نظر گرفتن شرایط اجرای محیط انجام می شود. ایجاد و مهاجرت نمونه VM، آخرین عملیاتی در فاز پیش مهاجرت است که ایجاد نمونه VM در دستگاه موبایل و ابر می کند.

عملیات حین مهاجرت با مهاجرت برنامه، در زمان اجرا و مهاجرت کلون VM سرو کار دارد. عملیات مهاجرت به فناوریهای شبکه بی سیم مربوط می شوند که به می توانند به صورت فناوریهای سلولی یا وای فای باشند. عملیات پس از مهاجرت دخیل در ایجاد نمونه VM در ابر، بوت اپ تصویر OS موبایل بر روی سرور ابر، شروع برنامه در ابر و حالات اجرای برنامه در ابر می باشند. فاز پس از مهاجرت، بستری را برای اجرای برنامه در ابر

فراهم می کند. فاز اجرا شامل پایش منبع در ابر است. منابع شامل سی پی یو، حافظه هستند. حسابداری منابع بر روی سیستمی سازی اطلاعات در خصوص استفاده از منابع در ابر انجام می شوند. حریم خصوصی در مرحله اجرای برنامه حاصل می شود. در نهایت، اجزای برنامه در طی مرحله حین اجرا هم زمان سازی می شوند. هم زمان سازی منبع دو نوع است 1- زمانی 2- رخداد محور. هم زمان سازی زمانی به طور دوره ای انجام می شود در حالی که هم زمان سازی برنامه مبتنی بر رخداد با تحریک رخداد انجام می شود. مرحله پس از اجرا شامل صرفه جویی در حالت های محلی و جهانی، مهاجرت نتیجه و ترکیب نتایج در خصوص ابزار موبایل است. حالت های محلی و جهانی برنامه در صورتی ذخیره می شوند که مهاجرت محاسبه نیاز باشد. در غیر این صورت نتایج حفظ شده و به دستگاه موبایل باز می گردد.

3-5 رده بندی چارچوب های اجرای برنامه

این بخش یک رده بندی کلی از چارچوب های اجرای برنامه را ارائه می کند. ویژگی های رده بندی، مربوط به خصوصیات چارچوب های اجرای برنامه می باشد که برای شرایط خودکار، کاربر شفاف، سازشی، چابک، قوی و زمینه آگاه برای دست یابی به اجرای برنامه های بهینه لازم است.

چارچوب ها برای پشتیبانی از پارتیشن بندی، افلود و ارائه خدمات مبتنی بر ابر خودکار مکانیسم ها در راستای دست یابی ب اجرای برنامه های بهینه لازم هستند. شفافیت کاربر را می توان با کمینه سازی مشارکت کاربر در فرایند اجرا بدست آورد. چارچوب اجرای کاربرد یا برنامه برای اجرای دو دستور لازم است. 1- مکانیسم تشخیص تغییر 2- مکانیسم پاسخ. این مکانیسم برای سخت، شفاف کاربر و چابکی نیز است. مکانیسم پاسخ باید موجب کاهش کیفیت برنامه و پاسخ به برنامه ها به شیوه ای فرصت طلبانه می شود. ویژگی چابکی را می توان با کمینه سازی سربار و تسریع فرایند محاسبه حاصل کرد. زمینه آگاهی به چارچوب امکان تشخیص فرصت های موجود را از حیث منابع و خدمات در رایانش ابری موبایل می دهد.

چارچوب های اجرای برنامه پیشرفته برای دست یابی به اهداف مختلف نظیر بهینه سازی هزینه اجراف کمینه سازی مصرف انرژی، کمینه سازی زمان انرژی و کمینه سازی تاخیر انرژی طراحی می شود. هم چنین رده بندی می تواند یک سری برنامه های موبایل مبتنی بر ابر نظیر یادگیری موبایل، سلامت موبایل، راهنمای موبایل، تشخیص چهره و واقعیت بزرگ نمایی موبایل ارائه کند. چارچوب های اجرای برنامه می توانند بر اساس نوع

پلتفرم طبقه بندی شود. استفاده از چارچوب بر سه نوع است 1- بر 2- تکه ابر و 3- هیبرید. ابر منابع و خدمات از راه دور ارایه می کند. اجرای برنامه منابع و سرویس های بیشتری را ارایه می کنند. اجرای برنامه در ابر از تاخیر WAN بالا رنج می برد که مانع از تحقق چشم انداز اجرای برنامه بهینه می شود. تکه ابر ها بر سه نوع می باشند. الف: مبتنی بر زیر ساخت ب: بدون زیر ساخت و ج: مبتنی بر زیر ساخت مجازی. تکه ابر مبتنی بر زیر ساخت نیازمند استفاده از سرور در WLAN و یا شبکه عرضه کننده سرویس ارتباط از راه دور است. تکه ابر بدون زیر ساخت نیازی به سرور برای توسعه تکه ابر ندارد بلکه از منبع دستکاه های موبایل موجود استفاده می کند. تکه ابر مبتنی بر زیر ساخت مجازی، مهاجرت برنامه به جای کل VM را انجام می دهد. برخی از اجزا از منابع دستکاه موبایل استفاده می کنند در حالی که باقی مانده از منابع ابری استفاده می کنند. پلتفرم هیبرید با ترکیب تکه ابر موبایل، تکه ابر محلی و از راه دور تشکیل می شود. هم چنین رده بندی چالش هایی را در تحقق دیدگاه اجرای برنامه نزدیک بهینه در MCC ارایه می کند.

6- شاخص های مهاجرت کاربرد بهینه در MCC

این بخش به بررسی مناسبت شاخص های مختلف مهاجرت برنامه موبایل می پردازد. شکل 5 شاخص ها را نشان می دهد. بررسی این شاخص ها در زمینه اجرای برنامه بهینه، یک بعد حقیقاتی مهم است. این شاخص ها مربوط به 5 زمینه شامل دستکاه موبایل، شبکه، نوع برنامه، ترجیح کاربر و هزینه است.

6-1 دستکاه موبایل

شاخص های مربوط به دستکاه موبایل از قابلیت های دستکاه از حیث بار پردازشی، سرعت سی پی یو، حافظه، ذخیره، فناوری های دسترسی بی سیم و تعداد رابط ها استفاده می کنند. بار پردازشی نشان دهنده بار کاری سی پی یو است در حالی که حافظه موجود و ذخیره اطلاعاتی را در مورد حافظه ارایه می کند و توسط برنامه استفاده می شوند. اگر بار پردازشی بالاتر باشد، فرایند برنامه جدید دارای تعداد کافی از سیکل های CPU برای اجرا مهم است. از این روی این برنامه نیازمند مهاجرت و سرور ابری برای اجرای یکنواخت است. حافظه موجود و حافظه مربوطه فضایی را برای حالت های برنامه و نتایج ارایه می کند. حافظه پایین منجر به اختلال در اجرای برنامه و ذخیره کم موجب خود داری از حفظ نتایج شود. از این روی حافظه کم مانع از اجرای برنامه شده و تجربه کاربر را کاهش می دهد. الگوی ابزار های موبایل بر روی خرابی لینک در شبکه موبایل تاثیر دارد. تصمیم افلود نیازمند

استفاده از الگوی تحرک است. فناوری دسترسی برای افلود برنامه استفاده می شود که بر عملکرد برنامه از حیث تاخیر و نسبت تحویل بسته تاثیر دارد. دستگاه موبایل با توان باتری پایین قادر به اجرای برنامه استفاده کننده از CPU با افلود به ابر است ضمن این که توان باتری را حفظ می کند. امروزه، دستگاه های موبایل از رابط های مختلف استفاده می کنند به طوری که افلود را می توان با استفاده از رابط های مختلف برای اجرای ترکیب پهنای باند بهبود بخشید. ترکیب پهنای باند به بهبود بازده، نسبت تحویل بسته و اطمینان پذیری کمک می کند. به علاوه، ترکیب پهنای باند موجب افزایش توان افلود برنامه می شود (رابمولی و همکاران 2012).



شکل 5: شاخص هایی برای مهاجرت برنامه بهینه در MCC

1- تینک ایر (ThinkAir) از مجازی سازی دستگاه موبایل برای اجرای هم زمان روش های افلود مختلف ریاضی استفاده کرده و از این روی زمان اجرای برنامه را کاهش می دهد. درخواست روش از راه دور از طریق کنترل گر اجرا صورت می گیرد و بر روی دستگاه موبایل و ابر قرار می گیرد. کنترل گر اجرا، تصمیم را بر اساس اطلاعات جمع آوری شده توسط نکارنده پروفیل می گیرد. در صورتی که ارتباط از راه دور با مشکل مواجه شود، چارچوب قادر به اجرای محلی نخواهد بود. سمت ابری کد افلود شده توسط سرور برنامه مدیریت می شود. چارچوب از شش

نوع VM با CPU متفاوت، و حافظه پشتیبانی می کند. سرور پیش فرض موسوم به سرور اولیه است و از این روی معمولاً آنلاین است و بقیه شامل نوع سرور ثانویه نیز هست. به علاوه چارچوب از پروفیل بندی صحیح و سبک وزن برای سخت افزار، نرم افزار و شبکه پشتیبانی می کند. چارچوب از اجرای موازی پشتیبانی کرده و سناریو های افلود چند کاربری را برای اجرا در ابر نظر گرفته می شود و از این روی چارچوب نیازمند نصب VM مبتنی بر سیستم عامل اندروید در سرور ابر می باشد.

6-2 نوع برنامه

ویژگی های برنامه موبایل از یک برنامه به برنامه دیگر متغیر هستند زیرا هر برنامه نیاز های مختلف را برآورده می کند. تصمیم افلود تحت تاثیر نوع برنامه است به خصوص اگر برنامه دارای پهنای باند بیشتری از نیاز های محاسباتی باشد. سپس بهتر است تا در مورد اجرای برنامه به طور محلی در دستگاه موبایل تصمیم گیری شود (هانگ و همکاران 2012). مطالعات زانک و فیگردیو (2006) برنامه های موبایل را به صورت سی پی یو محور، حافظه محور و ورودی-خروجی محور طبقه بندی می کند. دستگاه های موبایل نیز می توانند حساس به تاخیر (نازیر و همکاران 2009)، امنیت محور (کانو و دمونچ-اسننی 2011، خان و همکاران 2013، سوهاک و همکاران 2014) و شبکه محور (بالاگاس و همکاران 2007) باشند. برنامه های سی پی یو محور و حافظه محور از کاندید های مناسب برای افلود می باشند، اگرچه برنامه های شبکه، ورودی و امنیت محور دارای محدودیت هایی هستند که در ابر دور اجرا می شوند. با این حال، تصمیم گیری افلود برنامه حساس به تاخیر بر اساس تاخیر شبکه و زمان اجرا است. در صورتی که زمان اجرای برنامه طولانی تر از زمان اجرای دور باشد، آنگاه افلود مفید خواهد بود (ورلین و همکاران 2012 الف). در برخی از موارد، برنامه های کل افلود، عملکرد برنامه را تضعیف می کنند. این نوع برنامه ها دارای دو نوع مولفه قابل انتقال و غیر قابل انتقال هستند. بخش های غیر قابل انتقال می توانند به صورت یکی از موارد زیر باشند: 1- ماژولی که در ورودی و خروجی قرار دارد (کارو و همکاران 2010، عثمان و هالس 1998، او و همکاران 2006 الف)، 2- یک ماژولی که با سخت افزار دستگاه موبایل به صورت مستقیم تعامل دارد (عثمان و عیلس 1998)، 3- ماژولی که به کد بومی حساس است (گو و همکاران 2004). و یا ماژولی از برنامه که بر روی دستگاه موبایل قرار دارد (او و همکاران 2006 ب) و 4- ماژولی که مستقیماً از اطلاعات مربوط به دستگاه استفاده

می کند(گو و همکاران 2004). به طور خلاصه، برای دست یابی به اجرای برنامه بهینه در محیط رایانش ابری، یک سری دستورات باید در اغلب شرایط با در نظر گرفتن ویژگی ها اجرا شوند.

6-3 سلايق کاربر

برنامه های موبایل در محیط بی سیم پویا با طیف وسیعی از فناوری های دسترسی اجرا می شوند که در آن بین گزینه های موجود مختلف نظیر لینک پهنای باند بالا و هزینه توازن وجود دارد. کاربر نهایی، گزینه های موجود را برای دست یابی به اجرای بهینه برنامه اولویت بندی می کند ضمن این که محدودیت های مربوط به QoS، هزینه و امنیت را نیز بر طرف می کند. سلايق مرتبط با QoS توسط کاربر برای دست یابی به سطح کافی از کیفیت تجربه با در نظر گرفتن این بودجه تنظیم می شود. یک نسبت مستقیم بین QoS و هزینه پولی وجود دارد با این حال کاربر، QoS بالای را در هزینه پولی پایین تقاضا می کند. عرضه کننده های خدمات، نسبت های زیادی را برای سطوح مختلف QoS در نظر می گیرند. از این روی QoS یک پارامتر مهم برای اجرای برنامه بهینه یک برنامه بهینه سیار است. QoS موجب بهبود تاخیر کلی و بازده برای کاهش فاصله عملکرد بین اجرای محلی و دور می شود. مشابه با QoS، امنیت نیز موجب افزایش هزینه پولی کاربر می شود. در عین حال پیچیدگی پردازش و سربار اطلاعاتی را برای اجرای برنامه افزایش می دهند. افزایش پیچیدگی محاسباتی و سربار اطلاعاتی موجب افزایش تاخیر می شود به طوری که این موضوع مانع از تحقق اجرای برنامه بهینه در MCC می شود. از این روی ویژگی های مربوط به امنیت به خصوص احراز هویت و رمزگذاری برای در نظر گرفتن برنامه و نیاز های کاربر نهایی لازم هستند. هزینه بر اساس میزان خدمات و برای انواع مختلف شبکه ها متغیر است. گزینه های مرتبط با هزینه موجب تسهیل انتخاب خدمات مختلف با در نظر گرفتن بودجه می شوند. برای برنامه های نیازمند اجرای بهینه، کاربر می تواند سهم پهنای باند بیشتر بدست آورده و برای ترافیک در شبکه در اولویت قرار گیرد.

6-4: هزینه

هزینه معمولا شامل هزینه پولی شبکه های بی سیم و ابر است. شبکه های بی سیم نظیر لینک های تری جی برای کاربران موبایل دارای هزینه هستند در حالی که وای فای یک فناوری دسترسی رایگان است. کاربر می تواند از وای فای برای افلود برنامه و ترکیب نتایج استفاده کند تا هزینه کل کاهش یابد. در صورتی که وای فای موجود نباشد، کاربر موبایل به تری جی سویچ می کند. عرضه کننده های سرویس ابری مختلف همین سرویس را بر

روی سرعت های مختلف استفاده می کنند. کاربرد از ابر میان ابر های موجود با در نظر گرفتن نرخ و سرعت خدمات انتخاب می کند. هزینه ها برای منابع مختلف متغیر است از این روی کاربرد می تواند منابع بیشتری برای تسریع فرایند اجرا با انجام اجرا های موازی بدست آورد. برای دست یابی اجرای برنامه بهینه، کاربرد می تواند هزینه زیادی برای دسترسی و استفاده از منابع بیشتر بپردازد.

6-5 شبکه

پارامتر های مربوط به شبکه شامل تاخیر، کیفیت لینک بی سیم، پهنای باند موجود، امنیت و هزینه شبکه هستند. استفاده از این پارامتر ها در تصمیم افلود موجب کاهش تاخیر و ایجاد افلود با هزینه مطمئن می شود. انتخاب شبکه با تاخیر کم با کیفیت لینک بی سیم بهتر و پهنای باند موجود در دست یابی به اجرای برنامه های ابری با بهبود بازدهی و کاهش زمان اجرا مهم است.

7- تحقیقات آینده

هدف دست یابی به اجرای برنامه بهینه در MCC، زمینه را برای تحقیقات در ابعاد مختلف ارائه می کند. این ابعاد تحقیقاتی طراحی برنامه بهینه را پوشش می دهند و مستلزم طراحی چارچوب اجرای برنامه، طراحی و اجرای مکانیسم های تبادل اطلاعات، راه حل هایی برای توسعه اجزا، ایجاد برنامه کاربر شفاف، مدیریت زمان واقعی محیط ناهمگن، ارائه سرویس خودکار، و مقیاس پذیری راه حل ها است.

برنامه های سیار یا موبایل بر روی دستکاه های موبایل منبع محدود اجرا می شوند و از این روی این برنامه ها باید طوری طراحی شوند که موجب کاهش مصرف انرژی شوند، عملکرد را بهبود بخشند و نیاز به منابع بهتری داشته باشند. وابستگی میان ماژول های توزیعی برنامه باید در زمان طراحی در نظر گرفته شوند. به علاوه، طراحی چارچوب های اجرای برنامه که به اجرای برنامه در MCC حساس هستند، برای استفاده بهینه از منابع محدود دستکاه های موبایل اهمیت دارند. چارچوب های اجرای برنامه باید با در نظر گرفتن تکات کلیدی طراحی شوند الف: وابستگی در اجزای غیر تخصیصی باید کاهش یابد ب: استفاده از پهنای باند باید کاهش یابد پ: هزینه اجرایی باید کاهش یابد به علاوه، تغییرات بالا در دسترسی به پهنای باند بر شیوه تصمیم گیری در مورد چارچوب اجرایی اثر دارد و موقعیت اجرا را انتخاب می کند.

بهینه سازی برنامه و رفتار های چارچوب اجرایی با استفاده از برنامه و فنون زمینه آگاه مهم تر می شود. مکانیسم های جمع آوری اطلاعات زمینه ای سبک وزن برای جمع آوری اطلاعات مربوط به دستگاه موبایل و سرور ابر نیاز است. محیط MCC بسیار پویا می باشد از این روی اطلاعات زمینه ای برای استفاده از اطلاعات زمینه ای با تصمیمات افلود در نظر گرفته می شود. به علاوه، خدمات موبایل می توانند از اطلاعات زمینه ای 1- برای افزایش دقت باز یابی اطلاعات 2- کشف خدمات 3- استفاده از رابط ها و 4- ضمنی سازی تعامل کاربر استفاده می کند.

دیگر مسیر تحقیقاتی برای دست یابی به اجرای برنامه بهینه، یافتن یک مکانیسم استفاده از اجزای بهینه است. استفاده از اجزای بهینه اشاره به استفاده از راهبرد اجزا دارد که از طریق آن می توان اجزایی را با سر بار محاسباتی و ارتباطی اجرا کرد. برنامه های موبایل مبتنی بر ابر دارای اجزای مختلف بوده و این اجزا بر اساس خصوصیات و کارکرد های مربوطه متغیر است. یکی از اجزاء، رابطه کاربردی است که در ورودی دخیل است و اجزا باید بر روی دستگاه موبایل اجرا شود. اجزایی که با سخت افزار دستگاه موبایل تعامل دارند باید بر روی موبایل اجرا شوند. اجزای با وابستگی بالا به یک دیگر باید بر روی یک دستگاه اجرا شوند و وابستگی میان اجزای غیر تخصیصی باید کمینه سازی شود. از این روی نیاز به بررسی روش های ساده برای یافتن وابستگی میان اجزا برای زمان اجرا با سر بار پایین بر روی دستگاه موبایل می باشد.

برنامه موبایل در MCC می تواند بر روی هر پلتفرم، دستگاه موبایل، تکه ابر یا ابر بر اساس دسترسی به منابع در نزدیکی محلی یا دسترسی به ابر اجرا شود. اجرای برنامه از دستگاه موبایل به تکه ابر مهاجرت می کند یا سرور ابر در هر زمان اجرا می شود. مهاجرت برنامه از دستگاه موبایل تا تکه ابر یا ابر برای کاربر نیاز است و به این ترتیب طراحی چارچوب های اجرایی نیاز است که اجرای کاربر شفاف از برنامه در پلتفرم داشته باشند.

مدیریت زمان واقعی محیط ناهمگن در MCC اشاره به طراحی بهینه، پایش کارآمد و حفظ موثر محیط مشارکتی برای اجرای برنامه دارد. این نیاز از اهمیت زیادی در اجرای بهینه برنامه در MCC برخوردار است. مدیریت محیط رایانشی در MCC مستلزم پایش دارایی فیزیکی نظیر سرور، دارایی مهمی است که شامل VM، زیر ساختار شبکه نظیر صفحه کنترل سوئیچ و استفاده موثر از راهبرد های مدیریتی است (زکری و همکاران 2012). از این روی، نیاز به طراحی پایش موثر و مکانیسم حفاظتی موثر برای پایش محیط ابری با سر بار پایین است.

ارایه خدمات خودکار اشاره به خدمات مورد تقاضا با مشارکت خدمات کمینه نیازمند مدل سازی مدل پیش بینی عملکرد بهینه دارد که به طور دوره ای تقاضای آینده را پیش بینی می کند و منابع را تخصیص می دهد. یک نیاز مربوط به دستگاه موبایل است که، عملکرد برنامه را پیش بینی می کند، تقاضای آینده را برآورد می کند و منابع را به کاربران ابر با در نظر گرفتن نیاز های برنامه تخصیص می دهند.

8- چالش های آینده

در این بخش، ما به بررسی چالش های مهمی می پردازیم که مانع از بهینه سازی برنامه موبایل در MCC می شود. بحث مربوط به چالش های حل نشده مسیر های تحقیقاتی را برای محققان در زمینه تحقیقات آینده در MCC ارایه می کند.

8-1 کاربرد بهینه و طراحی چارچوب اجرایی

طراحی بهینه یک برنامه و چارچوب اجرای آن نیازمند تبادل داده ها، سر بار کم تر، فرایند مهاجرت سبک وزن و استفاده از مکانیسم های امنیت ناب است. پیچیدگی فرایند مهاجرت بالا است زیرا شرایط پویا و محیط MCC پیچیده از شبکه های بی سیم وجود دارد. طراحی برنامه بهینه و چارچوب اجرایی می تواند برنامه بهینه را برای محیط پویا و متنوع ارایه کند.

8-2 استفاده کارآمد و اجرای کاربر شفاف

استفاده کارآمد از برنامه مستلزم وابستگی کمینه در اجزای غیر عددی موجب کاهش فاصله تا منابع قابل دسترس، استفاده حداقل از پهنای باند، و هزینه اجرایی پایین می شود. با این حال، توسعه کارآمد برنامه در MCC به سختی به دلیل پیچیدگی تعیین وابستگی سخت است. به علاوه دست یابی به اجرای برنامه کاربر شفاف در MCC یک چالش تحقیقاتی باز به دلیل پیچیدگی در فرایند مهاجرت، محدودیت های ذاتی محیط بی سیم و ناهمگنی در پلتفرم MCC محسوب می شود.

8-3 مدیریت بهینه زمان واقعی محیط رایانشی همگن

مدیریت زمان واقعی محیط رایانشی ناهمگن به یک چالش به دلیل ماهیت متنوع عناصر محیطی، محدودیت های محیط بی سیم و الگو های تحرک پویا تبدیل شده است. با این حال، محققان می توانند دستور العمل هایی را از تلاش های قبلی در همین زمینه کسب کنند (پاونی و همکاران 2012) تا راه حل های ساده ای را برای

مدیریت زمان واقعی منابع ناهمگن طراحی کنند. کاهش تعداد عناصر پایشی و فشردگی مقدار تبادل اطلاعات برای مدیریت منابع در دست یابی به راهبرد مدیریتی زمان واقعی موثر است.

8-4-ارایه خدمات خودکار

هدف ارایه خدمات اتومات، ارایه منابع با کم ترین میزان خدمات می باشد. ارایه خدمات خودکار به دلیل پیچیدگی در مدل سازی مدل پیش بینی عملکرد سخت است. این را می توان با ترکیب رویکرد های مصنوعی برای پیش بینی عملکرد برنامه و تقاضای آینده انجام داد.

8-5-مقیاس پذیری

مقیاس پذیری موجب حصول اطمینان از ارایه خدمات صرف نظر از تعداد دستگاه ها با خدمات می شود. مقیاس پذیری دستگاه و ارایه منابع برای MCC در زمانی مناسب و مفید است که ابر غیر قابل دسترس باشد و تنها منابع محلی محدود قابل دسترس باشد. با منابع و خدمات محدود، اطمینان از مقیاس پذیری یک کار چالش بر انگیز است که بر اجرای برنامه اثر دارد. در این سناریو، مقیاس پذیری با اجرای وظایف اصلی زمینه ای بر روی دستگاه موبایل و اولویت بندی تخصیص منابع به وظایف انجام می شود.

8-6-قابلیت دسترسی به خدمات

قابلیت دسترسی به خدمات در ابر موجب حصول اطمینان از ارایه خدمات از راه دور می شود. قابلیت دسترسی خدمات ابری در MCC بستگی به سه عامل دارد. 1- محیط دسترسی به بی سیم 2- ظرفیت سرور باقی مانده و 3- تاخیر دسترسی به داده ها. چارچوب های اجرایی مبتنی بر MCC با جالش ها در اطمینان از دسترسی به خدمات ناشی از محدودیت های فناوری بی سیم و بار کاری سرور غیر قطعی ارتباط دارد. عدم توانایی در دسترسی به منابع کافی برای تقاضای برنامه منجر به اختلال در اجرای برنامه می شود. قابلیت دسترسی به منابع با مدیریت فعال منابع محلی برای اجرای برنامه در صورت کاهش از دست رفت ارتباط حاصل می شود.

9- نتیجه گیری

این مقاله به بررسی راهبرد های بهینه سازی پیچیده پیشرفته با ایجاد نظر سنجی در منابع می پردازد. این مقاله موارد زیر را در بر می گیرد 1- مرور منابع جامع 2- مقایسه چارچوب های اجرای برنامه پیشرفته 3- شناسایی راهبرد های بهینه سازی برنامه برای دست یابی به اجرای برنامه بهینه 4- طراحی رده بندی مربوط به برنامه 5-

بررسی شایستگی شاخص ها برای اجرای برنامه های بهینه 6- بررسی چالش های تحقیقاتی باز در دست یابی به اجرای برنامه بهینه. این مقایسه شان دهنده تفاوت ها و تشابهات در چارچوب های فعلی بوده و قابلیت چارچوب های موجود را برای اجرای بهینه برنامه ارزیابی می کند.

برخی از چارچوب های اجرای برنامه موجب بهبود زمان پاسخ برنامه برای MCC برای کاهش تاخیر WAN می شوند. تعدادی از این چارچوب ها فاقد مقیاس پذیری و پشتیبانی QoS هستند. به علاوه، برخی از چارچوب های فعلی نیازمند پشتیبانی برنامه نویس یا برنامه مبتنی بر VM می باشند. نیاز به طراحی شاخص های بهینه برای دست یابی به اجرای برنامه بهینه با در نظر گرفتن سلايق کاربر و محدودیت های بهینه وجود دارد. به علاوه، تحقیقاتی باید در حوزه چارچوب مورد بررسی با در نظر گرفتن معایب چارچوب های موجود انجام شوند.

این مقاله، از سری مقالات ترجمه شده رایگان سایت ترجمه فا میباشد که با فرمت PDF در اختیار شما عزیزان قرار گرفته است. در صورت تمایل میتوانید با کلیک بر روی دکمه های زیر از سایر مقالات نیز استفاده نمایید:

لیست مقالات ترجمه شده ✓

لیست مقالات ترجمه شده رایگان ✓

لیست جدیدترین مقالات انگلیسی ISI ✓

سایت ترجمه فا ؛ مرجع جدیدترین مقالات ترجمه شده از نشریات معتبر خارجی