EFICIENȚA ALGORITMILOR DE ALOCARE A RESURSELOR ÎN CLOUD COMPUTING

Ecaterina CIOBANU

Departamentul Inginerie Software și Automatică, grupa academică TI-231M, Facultatea de Calculatoare, Informatică și Microelectronică, Universitatea Tehnică, Chișinău, Republica Moldova

Autorul corespondent: Ecaterina CIOBANU, e-mail: ecaterina.ciobanu@isa.utm.md

Rezumat. *În contextul actual, expansiunea de amploare a cloud computing-ului marchează o evoluție semnificativă, amplificând astfel necesitatea unei gestionări eficiente a resurselor, o provocare ce influențează în mod direct performanța, costurile, scalabilitatea, fiabilitatea și sustenabilitatea mediului. Articolul prezent oferă o analiză exhaustivă asupra complexității algoritmilor de asignare a resurselor în cadrul mediilor cloud, făcând legătura în mod direct cu eficiența operațională, fiabilitatea sistemului, costurile și consumul de energie. Se investighează în profunzime algoritmii eficienți, cu scopul principal de a identifica și compara cele mai optime strategii pentru optimizarea distribuției resurselor în cloud computing. Se propune o clasificare a algoritmilor existenți, acoperind abordări euristice, deterministe și stocastice. Fiecare algoritm este supus unei evaluări amănunțite, fiind evidențiate atât punctele lor forte, cât și limitările pe care le posedă. Aceste informații esențiale vizează îmbunătățirea performanței algoritmilor de alocare a resurselor, sporind astfel eficiența, scalabilitatea, fiabilitatea și rentabilitatea serviciilor cloud.*

Cuvinte cheie: *algoritmi euristici, algoritmi deterministici, algoritmi stocastici, scalabilitate, fiabilitate*

Introducere

Computing cloud este un model de serviciu al cărui idee de bază este oferirea la cererea a resurselor computaționale partajate. Cloud computing reprezintă o paradigmă revoluționară de în calculul distribuit pe scară largă. În acest model, resursele de calcul sunt puse la dispoziție clienților externi, la cerere prin intermediul internetului. Printre aceste resurse se numără: spațiul de stocare, platforme și servicii abstractizate, virtualizate și scalabile dinamic. Această abordare permite accesul facil la resursele de calcul, fără a fi necesar pentru client să investească în infrastructura proprie. Aceste tehnici au fost în vogă doar pentru o perioadă și nici una dintre ele nu a fost la fel de influentă ca computing cloud. Odată cu disponibilitate mai mare a internetului și a resurselor informatice, companiile mari au început adaptarea acestui model nou de servicii. Astfel, companiile principale pe piața internetului, au adaptat acest model de business, Amazon cu Amazon Web Services(AWS), Microfost cu Microsoft Azure și Google cu Google Cloud Platform(GCP).

Cloud computing este definit prin câteva caracteristici esențiale, printre care elasticitatea și disponibilitatea. Elasticitatea permite ajustarea dinamică a resurselor în funcție de cerințe, optimizând utilizarea și reducând costurile[1]. Disponibilitatea asigură accesul fără întreruperi semnificative, prin utilizarea de infrastructuri redundante și planuri de backup.

Serviciile de Cloud Computing sunt împărțite în următoarele categorii:

- Infrastructure as a Service (IaaS) - furnizează resurse IT um ar fi serverele, stocarea și rețelele la cerere, clienții având controlul deplin asupra aplicațiilor și sistemelor instalate,
- Platform as a Service (PaaS) - oferă platforme de dezvoltare și implementare a aplicațiilor, utilizatorii pot dezvolta și implementa aplicații pe această platformă fără a fi nevoiți să își gestioneze sau să își configureze infrastructura subiacentă.

- Software as a Service (SaaS) – furnizează aplicații software accesibile prin internet, utilizatorii pot accesa aceste aplicații direct prin intermediul unui browser web, fără a fi nevoie să instaleze sau să gestioneze software-ul pe propriile dispozitive.

Alocarea corectă a resurselor și utilizarea algoritmilor potriviți au un impact direct asupra performanței cloud-ului, eficienței costurilor și asupra eficienței cloud-ului în general. Mediul cloud este foarte dinamic și scalabil ceea ce presupune că trebuie să fie ajustabile la sarcini variabile și cerințe variate in timp real.

Este important să fie corect alocate resursele din următoarele considerente:

- Optimizarea performanței, asigură accesul la resursele necesare, sporind astfel performanța și satisfacția utilizatorilor,
- Scalabilitate, permite serviciilor sa gestioneze eficient sarcinile de vârf și să economisească resurse în perioade mai puțin aglomerate,
- Securitate, asigurară că sarcinile de lucru sunt distribuite în conformitate cu cerințele legale și de reglementare, care pot include considerente de rezidență și protecție a datelor.

Algoritmii utilizați pentru alocarea resurselor

Cele mai frecvente tipuri de algoritmi utilizați:

1. Algoritmii euristici (algoritmii min-min si min-max);
2. Algoritmi deterministici (First come first serve, Round Robin),
3. Algoritmi stocastici (algoritmul genetic, Ant Colony Optimization),

Algoritmii euristici

Euristica presupune o tehnica de rezolvare a problemelor mai rapida decat ar putea propune alte tehnici. Acest tip de algoritmi este atribuit algoritmilor de aproximare si are ca scop găsirea unei soluții particulare rezonabile in timp. În continuare vor fi analizați algoritmii MIN-MIN și MAX-MIN.

Algoritmul MIN-MIN

Acest algoritm începe cu un set (notat cu U) de sarcini nemapate. Se alege un subset M de sarcini cu cel mai mic timp necesar pentru completare, aceste sarcini se alocă unei mașini disponibile și se șterg din U. Se repetă din nou procedura cu alegerea task-urilor cu cel mai puțin timp pentru compilare din setul U și după sunt alocate resurse pentru efectuarea lor. Algoritmul este stopat când mulțimea U devine vidă. Principala problemă a algoritmului este că se acordă prioritate sarcinilor mici, fiind crescut timpul de finalizare pentru sarcinile mai complexe și mari.

Algoritmul MAX-MIN

Acest algoritm de asemenea începe de la un se U de task-uri. Se alege un subset M care conține task-uri cu cel mai mic timp de execuție. Din subsetul obținut se alege task-ul cu cel mai mare timp de utilizare și este alocat la cea mai puternică mașină virtuală disponibilă. Algoritmul continuă până când setul U devine gol. Algoritmul MAX-MIN ajută la îmbunătățirea intervalului de execuție a setului de sarcini, alocând task-urile complexe la cele mai performante mașini virtuale.

Algoritmii deterministici

Algoritmii deterministici sunt algoritmii care pentru aceleași date de intrare produc aceleași date de ieșire de fiecare dată când sunt executați. Astfel asigură predictibilitate și fiabilitate în diferite sarcini de calcul. Acest tip de algoritm este potrivit pentru procesele de sistem unde precizia și stabilitatea sunt primordiale.

First come first Serve(FCFS)

FCFS este un algoritm de planificare și alocare a resurselor unor task-uri sau job-uri. Aceste resurse includ timpul CPU, memoria, lățimea de bandă a rețelei, care sunt de regulă încapsulate în mașina virtuală(VM) utilizată.

FCFS este cel mai simplu algoritm de alocarea a resurselor, se bazează pe o singură regulă: alocă resurse pentru primul proces din coadă și îl lasă să ruleze până la final. Este un algoritm non-preemptiv, ceea ce presupune că poate rula doar un proces la un moment dat, indiferent dacă există o coadă de alte procese cu importanță mai ridicată [2]. Din cauza acestor limitări acest algoritm nu este folosit pe scară largă.

Round Robin

Round robin este un algoritm vechi și destul de simplu și corect de planificare a resurselor. Se definește o mică unitate de timp, numită quantum. Toate procesele rulabile se află într-o coadă circulară. Planificatorul CPU parcurge această coadă, alocând CPU-ul pentru fiecare proces, timp de un quantum. Dacă procesul nu s-a terminat până la sfârșitul quantum-ului, acesta este adăugat la sfârșitul cozii, iar resursele CPU trec la următorul proces. Dacă procesul se sfârșește înainte de a expira timpul din quantum, resursele CPU sunt automat eliberate și trec la următorul proces. În cazul în care apare un proces nou, acesta este adăugat la sfârșitul cozii.

Avantajul acestui algoritm este că un job nu trebuie să aștepte să fie finalizat precedentul. De fiecare dată când unui proces i se acordă resurse, are loc o schimbare de context, ceea ce adaugă timp extra la timpul total de execuție.

Algoritmii stocastici

Optimizarea stocastică urmărește să atingă soluții adecvate la probleme multiple, similare optimizării deterministe. Totuși, diferit de optimizarea deterministă, algoritmii de optimizare stocastică folosesc procese cu factori aleatorii pentru a face acest lucru.

Datorită acestor procese cu factori aleatorii, optimizarea stocastică nu garantează găsirea rezultatului optim pentru o anumită problemă. Dar, există întotdeauna o probabilitate de a găsi rezultatul optim la nivel global [3].

Algoritmul genetic

Algoritmul genetic este un algoritm pentru optimizarea algoritmilor care imită procesele evolutive. În contextul alocării resurselor de cloud computing, algoritmii genetici sunt utilizați pentru a optimiza alocarea resurselor cum ar fi CPU, memorie, stocare și lățime de bandă a rețelei între diferite sarcini sau mașini virtuale (VM) pentru a atinge obiective specifice, cum ar fi minimizarea costurilor, maximizarea performanță sau echilibrarea sarcinii între resurse.

Studiu de caz

Cercetările recente în domeniul programării resurselor în cloud au explorat diverse tehnici de optimizare pentru îmbunătățirea eficienței. Alkayal și colegii au propus un algoritm de optimizare a roiului de particule (PSO) pentru prioritizarea și asignarea sarcinilor pe mașini virtuale în funcție de durată. Mezmaza și echipa sa au utilizat un algoritm genetic hibrid paralel pentru găsirea setului optim de sarcini, cu accent pe eficiența energetică și reducerea timpului de procesare. Mocanu și colaboratorii au introdus un algoritm genetic care utilizează roata ruletei și se concentrează pe minimizarea timpului de execuție, cu o funcție de fitness axată pe utilizare. Geetha și colegii lor au combinat rețele neuronale și algoritmi genetici pentru a gestiona cererile nelimitate într-un sistem paralel și distribuit, cu atenție și la un cloud federat. Zhou și echipa sa au prezentat un algoritm genetic extensibil care integrează căutarea locală bazată pe heuristici și o etapă de creștere pentru a permite indivizilor să evolueze prin rute diferite de creștere. În fine, Ajak și colaboratorii săi au dezvoltat un model de programare bazat pe grafuri aciclice dirijate (DAG) pentru optimizarea timpului total de procesare prin alocarea eficientă a sarcinilor și organizarea secvenței de execuție, folosind tehnici euristice și de aprovizionare a resurselor.

Ant Colony optimization Algorithm

Ant colony algorithm (ACO), inspirat de comportamentul furnicilor, este o tehnică probabilistică pentru rezolvarea problemelor de calcul care poate fi redusă la găsirea de căi bune prin grafice. În contextul cloud computing-ului, ACO poate fi aplicat problemei alocării și programării resurselor. Aceasta implică alocarea resurselor de cloud computing, cum ar fi puterea de calcul și stocarea, pentru diverse sarcini într-un mod eficient pentru a optimiza anumite obiective, cum ar fi minimizarea costurilor, consumului de energie sau timpul de execuție, menținând în același timp un nivel ridicat de performanță a serviciilor.

Ideea de bază constă în faptul că este un nod master, responsabil de distribuirea sarcinilor și a resurselor mașinilor virtuale. Dacă execuția task-ului eșuează el este întors către nodul master, și atribuit unei mașini virtuale mai puternice. Dacă se atestă faptul că o mașină virtuală nu se descurcă cu sarcina atribuită, sau resursele ei sunt la maxim utilizate, atunci nodul master poate aloca resurse adăugătoare pentru a ajuta la execuția lui [4].

Acest algoritm este util în mediile de cloud computing datorită flexibilității, scalabilității și capacității lor de a găsi soluții bune în scenarii complexe și dinamice în care tehnicile tradiționale de optimizare ar putea avea dificultăți. S-a dovedit a fi eficace în îmbunătățirea eficienței și a receptivității serviciilor cloud, contribuind la practici de cloud computing mai sustenabile și mai rentabile.

Analiza comparativă a algoritmilor

În continuare se propune analiza comparativă a acestor algoritmi conform criteriilor: scalabilitate, corectitudine, eficiența costurilor, fiabilitate, practicabilitate, complexitate. Astfel, poate fi observat că fiecare categorie de algoritmi are puncte forte și puncte slabe. Din punct de vedere al scalabilității, algoritmii deterministici sunt mai puțin eficienți. Din punct de vedere al costurilor, algormitmii stocastici sunt scumpi inițial, dar pe termen lung își răscumpără costul, în timp ce algoritmii euristici și deterministici sunt eficienți la acest capitol. La punctul fiabilitate, cel mai fiabil algoritm este cel deterministic, după care urmează cel euristic, care poate fi imprevizibil în unele cazuri, după care cei stocastici, care pot necesita resurse în plus.

Compararea între categoriile de algoritmi este realizată în tabelul 1.

Tabelul 1

	Algoritmii euristici	**Algoritmii deterministici**	**Algoritmii stocastici**
Scalabilitate	destul de bună	restrânsă	destul de bună
Corectitudine	nu este garantată	trebuie programată	variază
Eficiența costurilor	eficient	Poate fi eficient	Inițial scump, dar în termen lung efectiv
Fiabilitate	Variază, în unele scenarii este imprevizibil	Foarte fiabil	Fiabil, dar uneori necesită resurse în plus
Practicabilitate	practic	practic	Practic pentru soluții complexe
Complexitate	Complexitatea variază	complex	complex

Concluzie

În acest articol a fost efectuată o analiză comparativă a algoritmilor, care oferă o perspectivă detaliată asupra modului în care diferitele categorii de algoritmi influențează eficiența și performanța în gestionarea resurselor în mediul cloud computing.

Astfel, abordarea adecvată a algoritmilor de gestionare a resurselor în cloud computing este esențială pentru îmbunătățirea performanței, eficienței și costurilor serviciilor cloud.

Algoritmii euristici oferă o soluție rapidă și aproximativă în rezolvarea problemelor complexe, în timp ce algoritmii deterministici asigură o abordare predictibilă și stabilă. În același timp, algoritmii stocastici aduc o perspectivă probabilistică și pot fi eficienți în gestionarea sarcinilor complexe și variabile.

Prin înțelegerea și utilizarea adecvată a fiecărei categorii de algoritmi, organizațiile pot optimiza utilizarea resurselor și pot îmbunătăți serviciile oferite clienților în mediul cloud. Adaptarea soluțiilor în funcție de nevoile specifice și obiectivele organizației poate contribui la maximizarea eficienței operaționale și a satisfacției clienților în mediul dinamic al cloud computing.

Referință

[1] Mahdi Manavi, Yunpeng Zhang, Guoning Chen, Resource Allocation in Cloud Computing Using Genetic Algorithm and Neural Network Houston, USA, 22 august 2023

[2] Artan Mazrekaj, Dorian Minarolli, Distributed Resource Allocation in Cloud Computing Using Multi-Agent Systems of Telfor Journal, Vol. 9, No. 2, 2017

[3] Karlisa Priandana, Modification of the Ant Colony Optimization Algorithm for Solving Multi-Agent Task Allocation Problem in Agricultural Application of Journal of Advanced Research in Applied Sciences and Engineering Technology · November 2023

[4] Guilherme Thompson, Stochastic models for resource allocation in large distributed systems of Sciences Mathématiques de Paris Centre, ED 386, 8 Oct 2018