



UNIVERSITAT
ROVIRA I VIRGILI



SISTEMES DISTRIBUÏTS

Task 1

EQUIP:	Adrià Cusidó Mas Miquel Buxons Vives
DATA:	15 d'Abril del 2020

Índex

1. Explicació De la solució.....	3
Main	3
Generador de matrius	3
Generador de l'iterdata	3
Funció map	3
Funció Reduce	4
2. Speedup i tests utilitzats	4
3. Distribució de les tasques	6
4. Bibliografia.....	6

1. Explicació De la solució

En aquest apartat realitzarem un breu explicació de cada tros de codi (molt breument) amb petites particularitats que pugui tenir cada segment.

Main

El segment principal s'encarrega de realitzar les crides al generador de matrius a partir de la seves dimensions, es crida una vegada per cada matriu, posteriorment crida al generador del vector "iterdata" (l'input per la funció map_reduce()) i realitza la crida a la funció map_reduce(), finalment realitza l'escriptura per pantalla de les dades més importants.

Generador de matrius

En aquesta funció es crea una matriu de les dimensions passades per paràmetre i amb nombres definits amb un rang que està definit com a global, posteriorment es desa la matriu completa al núvol (serialitzada) i es procedeix a realitzar la partició en files o columnes i el consegüent emmagatzemament al núvol de cada tros per separat i segons el seu nom, també passat per paràmetre.

Generador de l'iterdata

En aquesta funció es realitza la repartició de feina entre cada Worker existent (el nombre de workers estarà limitat entre 1 i el nombre total de chunks o 100 en el cas de matrius grans). Aquesta funció genera un vector general en que cada posició serà un altre vector que contindrà els identificadors de les files i columnes que ha de multiplicar cada Worker.

Funció map

Aquesta funció rep un vector en que cada posició és un altre vector fix de dues posicions que pertanyen als dos identificadors (un per la fila de la matriu A i l'altre per la columna de la matriu B) que es volen multiplicar i sumar posteriorment. Es realitza la descarrega des del bucket del núvol, es realitza la suma vectorial i amb el valor obtingut, juntament amb el numero de la fila i la columna que pertany s'envien a la funció reduce.

Funció Reduce

Aquesta funció rep els vectors en que cada posició és un altre vector amb el format (fila, columna, valor) pertanyents a cada chunk i que envia cada worker, aquesta funció realitza la inserció de cada element a la seva posició de la matriu i finalment emmagatzema la matriu resultat sencera al núvol.

2. Speedup i tests utilitzats

Hem dut a terme una sèrie de proves per comprovar com afectava la paral·lelització de les tasques en el temps total d'execució i en el rendiment.

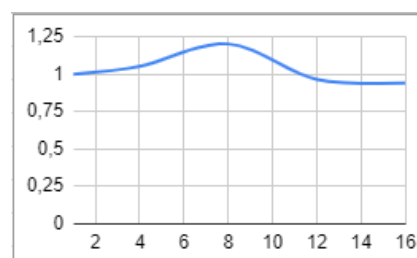
Hem començat multiplicant matrius petites que de manera seqüencial tenien un temps d'execució bastant reduït. Hem observat que si volíem utilitzar un nombre massa gran de workers, sempre utilitzant com a màxim el número de multiplicacions de files per columnes, el rendiment i el temps d'execució no només no millorava, sinó que fins i tot empitjorava bastant. Aquest fenomen pot ser explicat perquè dividir el codi en fragments tant petits i senzills per executar-los al cloud provoca un overhead més gran que l'execució seqüencial del mateix. Això sí, abans d'aquest decrement en el rendiment, sempre hi havia un augment utilitzant un número no massa elevat de workers.

A mesura que la mida de les matrius augmentava, també ho feia el número de workers que podíem utilitzar per aconseguir que el temps d'execució fos inferior a l'obtingut de manera seqüencial.

A les proves amb matrius que ja començaven a ser d'una mida bastant elevada hem observat com el rendiment i el temps d'execució milloraven de forma considerable fins i tot utilitzant un número no massa elevat de workers.

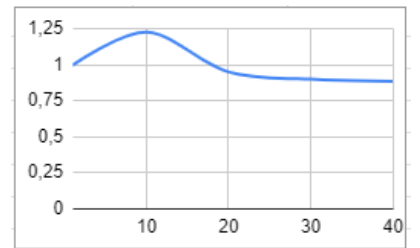
Matrius quadrades de 4x4

4x4 * 4x4	seq	4	8	12	16
	3,416658401	3,669820786	2,956502438	3,888896465	3,853856087
	3,865664244	3,210636854	3,074023247	3,757567167	3,79325223
	3,505039215	3,304350615	3,219791174	3,51084137	3,241370678
	3,640228987	3,176954269	2,660661221	3,830522776	4,524749041
	3,395447493	3,562525511	2,886621237	3,461633205	3,512022257
Mitjana	3,564607668	3,384857607	2,959519863	3,689892197	3,785050058
Speedup	1	1,053	1,204	0,966	0,942



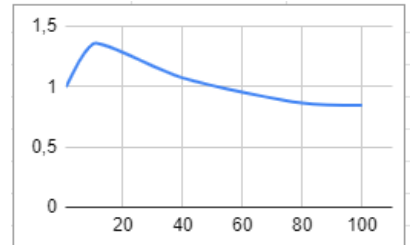
Matriu de 8x4 i matriu de 4x5

8x4 * 4x5	seq	10	20	30	40
	3,5030334	3,311353207	4,21554637	4,253937483	4,385946989
	3,53550005	3,513720036	4,08239603	4,195438623	4,040561676
	4,318257332	3,344760656	4,450515985	4,629311085	4,351258278
	4,854356527	3,062483311	4,461448431	5,210197687	5,13006115
	4,295860291	3,436995983	4,350330114	4,474437475	5,231553793
Mitjana	4,10140152	3,333862638	4,312047386	4,552664471	4,627876377
Speedup	1	1,230	0,951	0,901	0,886



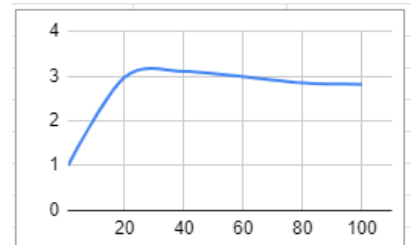
Matrius quadrades de 10x10

10x10 * 10x10	seq	10	40	80	100
	5,008352518	3,834718704	4,85256815	5,218800783	5,773338556
	4,735057116	3,515046835	4,752310514	5,816993237	5,715368748
	4,581214905	3,790765047	4,556352615	5,978102207	5,840396404
	5,068376541	3,287534952	4,387789726	5,820160627	5,935261965
	5,08011508	3,593028069	4,225423336	5,504131556	5,661222219
Mitjana	4,894623232	3,604218721	4,554888868	5,667637682	5,785117579
Speedup	1	1,358	1,075	0,864	0,846



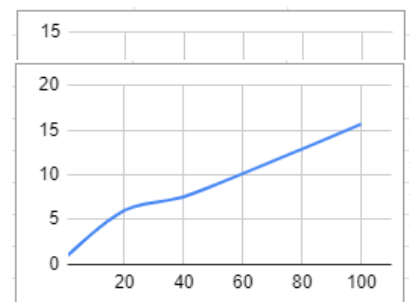
Matriu de 20x35 i matriu de 35x40

20x35 * 35x40	seq	20	40	80	100
	19,12881613	5,92377305	6,108769178	7,001621723	6,334892035
	18,57441092	6,845803976	5,772005796	6,982293844	7,010561466
	19,3384285	6,955176592	6,207442045	6,788300037	6,129961967
	18,31142926	6,18042469	6,39055109	6,493043661	7,180862904
	19,46337891	5,949880362	6,010546446	6,012958765	6,947843313
Mitjana	18,96329274	6,371011734	6,097862911	6,655643606	6,720824337
Speedup	1	2,976	3,110	2,849	2,822



Matrius quadrades de 100x100

100x100 * 100x100	seq	20	40	80	100
	204,1533051	37,55231619	28,96289659	19,4290905	17,19039297
150x150 * 150x150	seq	20	40	80	100
	455,9570947	75,45179796	61,19052887	35,5028882	27,12132668
	455,0665214	76,00953102	61,36514616	38,55762625	30,01789236
	456,3684653	75,79528975	60,02287436	34,76607823	30,4444828
	454,5373756	75,49870896	59,99984241	33,13450527	28,83993793
	455,0170928	75,7722323	59,70175791	34,69902968	28,57228208
Mitjana	455,3893099	75,70551019	60,45602994	35,33202553	28,99918437
Speedup	1	6,015	7,533	12,889	15,704



Matrius quadrades de 150x150

3. Distribució de les tasques

Per realitzar la pràctica hem procedit de la següent manera, primerament vam construir una part seqüencial sense execució al núvol, després vam anar aplicant refinacions al algoritme afegint-hi l'execució al núvol (seqüencial), paral·lelisme, nombre de workers variables i diferents dimensions per cada matriu. Per tant, en el programa principal com en les funcions de Map i Reduce els dos membres del grup vam treballar en conjunt i seria impossible distingir la feina de l'un i de l'altre.

Per altra banda, si que podríem distingir com a autor de la funció de generació i emmagatzemament de la matriu a l'Adrià Cusidó i la funció de creació del vector Iterdata al Miquel Buxons.

4. Bibliografia

IBM - PyWren - PyWren on IBM Cloud, URL: <https://github.com/pywren/pywren-ibm-cloud>

Numpy - documentation, URL: <https://docs.scipy.org/doc/numpy/reference/>

Numpy - dot, URL: <https://docs.scipy.org/doc/numpy/reference/generated/numpy.dot.html>