



Vectorización de matrices

Tiempos de ejecución de la multiplicación
de matrices convencionales y vectorizadas

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Computación III

Lic. en Tecnología

Miguel Alejandro López Olvera

20-11-2020

Introducción

En este documento se muestran comparaciones del tiempo de ejecución de multiplicaciones de matrices convencionales y vectorizadas. El código se implementó en Python y es capaz de realizar sumas, restas y multiplicaciones de matrices convencionales y vectorizadas, además de tomar sus tiempos de ejecución, pero para el objetivo de este reporte solo se comparan los tiempos de las multiplicaciones, además de que se genera un gráfico de línea donde se muestra el comportamiento del SpeedUp.

Desarrollo

La multiplicación de matrices convencionales se ejecuta con ayuda del siguiente código, el cuál fue implementado en Python y se puede encontrar el método completo en el enlace adjunto en el apéndice A.

```
for i in range(len(self.M_2)):
    for j in range(len(self.M[0])):
        for k in range(len(self.M)):
            z[i][j] += self.M[i][k]*self.M_2[k][j]
```

Fig.1 Parte del código encargada de realizar la multiplicación convencional

Para la multiplicación de las matrices vectorizadas fue necesaria la implementación de otro código que de igual forma se puede encontrar en el enlace adjunto en el apéndice A.

```
a=0
b=0
for i in range(len(V_2)):
    for j in range(self.c):
        z[i] += V[j+a]*V_2[b+j*self.c]
        if((j+1)%self.c == 0):
            b+=1
    if((i+1)%self.c==0):
        a+=self.c
        b=0
```

Fig. 2 Parte del código encargada de realizar la multiplicación vectorizada

Resultados

Se tomaron 2 tiempos de ejecución de 20 y 10 multiplicaciones de matrices convencionales y vectorizadas, obteniendo las siguientes matrices de los tiempos de ejecución, además de un SpeedUp.

```
Tiempos, multiplicación convencional:
[2.00271606e-01 1.31034851e+00 4.37474251e+00 1.03905201e+01
 2.04069614e+01 3.45218182e+01 5.38203716e+01 8.37256908e+01
 1.15075111e+02 1.76675797e+02 2.14846373e+02 2.70737410e+02
 3.52186203e+02 4.31198359e+02 5.31570911e+02 6.40740395e+02
 7.68358231e+02 9.33158875e+02 1.07850218e+03 1.28024888e+03]
Tiempos, multiplicación vectorizada:
[1.49011612e-01 1.22475624e+00 3.10373306e+00 7.28106499e+00
 1.46820545e+01 2.49860287e+01 4.02510166e+01 5.95932007e+01
 8.59067440e+01 1.23027086e+02 1.57470942e+02 2.03187704e+02
 2.60719061e+02 3.31016302e+02 4.02461767e+02 4.91295338e+02
 5.88493824e+02 6.97026253e+02 8.11716080e+02 9.59896803e+02]
SpeedUP:
[1.344      1.06988515 1.40950991 1.42706048 1.38992546 1.38164486
 1.33711832 1.40495375 1.33953525 1.43607235 1.36435567 1.33244977
 1.35082645 1.30264992 1.32079853 1.30418578 1.30563516 1.33877149
 1.32866923 1.33373596]
```

```
Tiempos, multiplicación convencional:
[1.24110603e+03 9.82881665e+03 3.42364419e+04 7.97154016e+04
 1.56225754e+05 2.67670723e+05 4.29999188e+05 6.43663533e+05
 9.11387299e+05 1.24766855e+06]
Tiempos, multiplicación vectorizada:
[9.17799473e+02 7.32589936e+03 2.92687562e+04 5.99880013e+04
 1.17901895e+05 2.03460929e+05 3.27123083e+05 4.91964165e+05
 6.95218775e+05 9.58355947e+05]
SpeedUP:
[1.35226274 1.34165324 1.16972657 1.32885577 1.32504871 1.31558784
 1.31448745 1.30835451 1.31093597 1.30188429]
```

Fig. 3 Tiempos de ejecución de la multiplicación de matrices y SpeedUp obtenido.

A continuación, se muestran dos gráficos de línea del SpeedUp encontrado para la multiplicación de 20 y 10 matrices, con dimensiones de 5 a 100 por lado y de 100 a 1000, respectivamente.

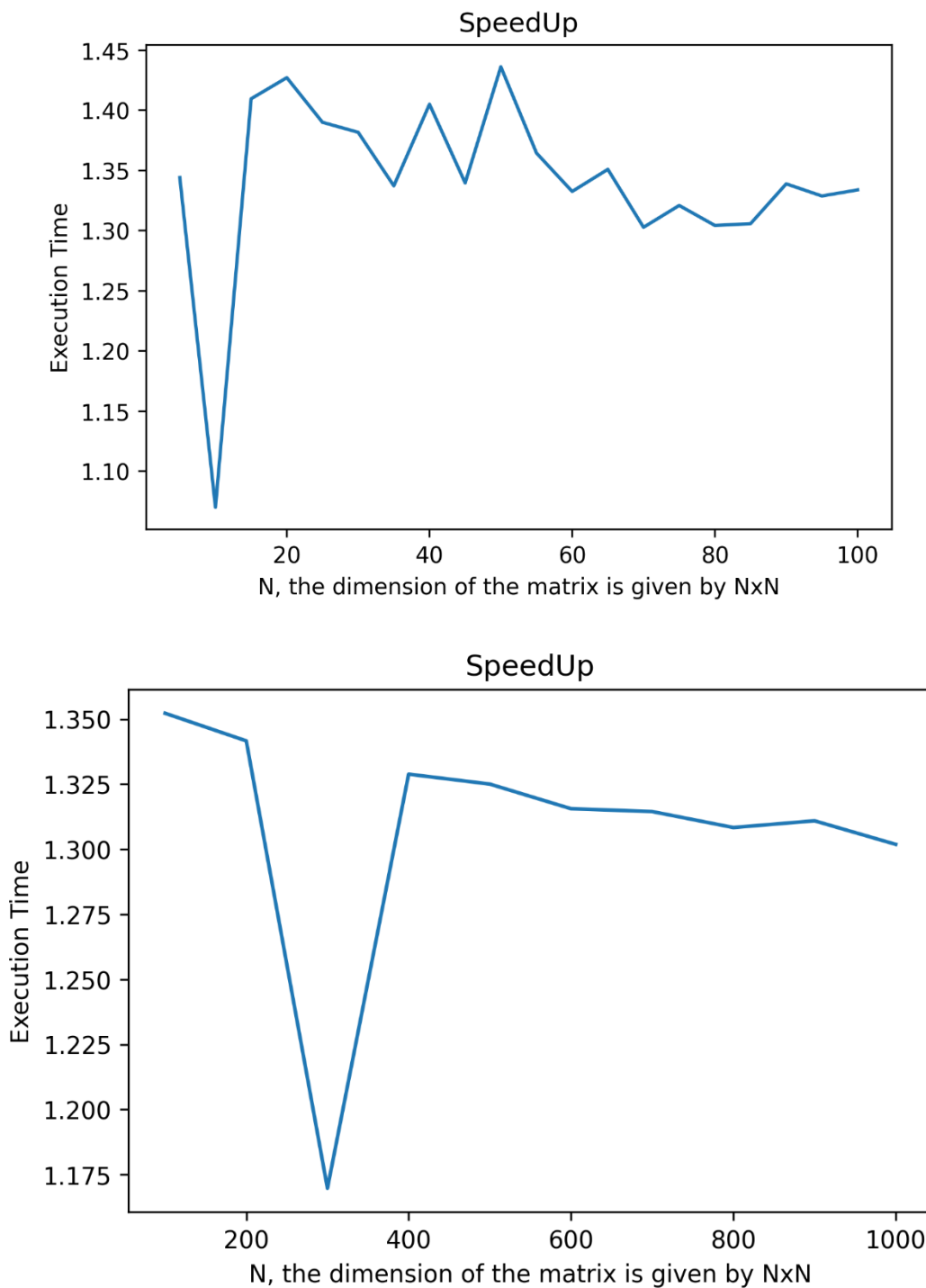


Fig. 4 Gráfico de línea que muestra el comportamiento del SpeedUp.

Conclusiones

Se puede apreciar que existe una tendencia de disminución del SpeedUp (relación del tiempo de ejecución de la multiplicación de matrices convencionales vs vectorizadas) conforme se aumenta el tamaño de la matriz. No fue posible realizar la multiplicación de matrices de mayor tamaño debido al elevado tiempo de ejecución, pero con los datos obtenidos para matrices de 5x5 a 100x100 y de 100x100 a 1000x1000 se puede apreciar el comportamiento del SpeedUp. No está de más mencionar que la ejecución de las multiplicaciones de las matrices vectorizadas resulta más eficiente que la multiplicación de matrices convencionales.

Apéndice A

Enlace del código en Google Colab:

<https://colab.research.google.com/drive/188FKjdNFRZdP3h5JPfHVx7dpl6qjmxKn#scrollTo=Wa2rNanjcRPE>