

Multiplicación de Matrices Secuencial/Paralela  
CUDA C

Presentado por:  
CHRISTIAN CAMILO RESTREPO LEDESMA

UNIVERSIDAD TECNOLÓGICA DE PEREIRA

Presentado a:  
John Osorio

Septiembre 30 de 2015  
PEREIRA

## INTRODUCCIÓN

El siguiente informe trata de ilustrar como es el comportamiento de la multiplicación de matrices implementada de diferentes maneras, la cual indicara cual es la manera más óptima.

Se tomaron diferentes tamaños para la implementación de la multiplicación con el fin de identificar en que situación y que algoritmos son los recomendables para un mejor desempeño.

Las pruebas se realizaron en la plataforma la cual compila y ejecuta los algoritmos en CUDA C.

## DESARROLLO

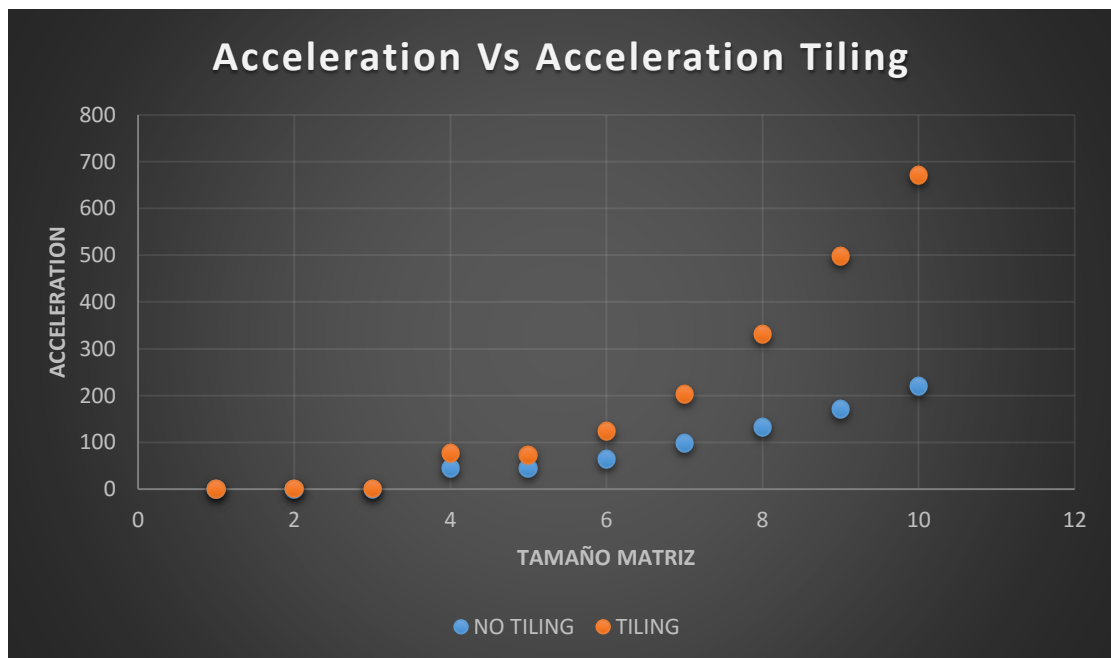
Los puntos 1, 2 se encuentran alojados en el repositorio

<https://github.com/ckrestrepo/HPC/Parcial>

3. La siguiente tabla ilustra los datos tomados para números enteros con un BlockSize de 32 y un TILE\_WIDTH de 32.

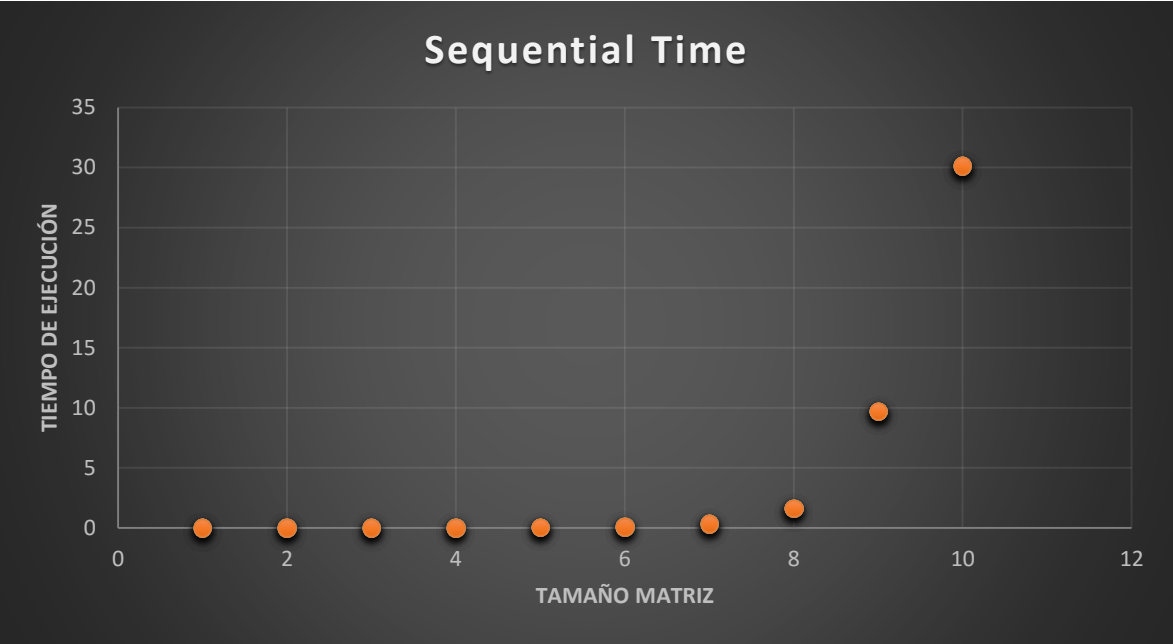
BlockSize = 32			Time				
Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	12x16	16x32	0,000017	0,000066	0,000004	0,257575758	0,425
2	32x64	64x10	0,000057	0,000088	0,000046	0,647727273	1,239130435
3	48x30	30x18	0,000075	0,000085	0,000048	0,882352941	1,5625
4	128x128	128x256	0,013376	0,000298	0,000173	44,88590604	77,31791908
5	256x128	128x160	0,014088	0,000314	0,000193	44,86624204	72,99481865
6	320x256	256x240	0,052593	0,000808	0,000424	65,09034653	124,0400943
7	512x360	360x600	0,329217	0,003337	0,001618	98,65657776	203,4715698
8	1024x512	512x960	1,635663	0,01231	0,004929	132,8727051	331,8447961
9	2048x1024	1024x1200	9,693118	0,056372	0,019458	171,9491592	498,1559256
10	2048x2048	2048x1500	30,109003	0,136082	0,044806	221,2563234	671,9859617

## 4. Grafica de aceleración

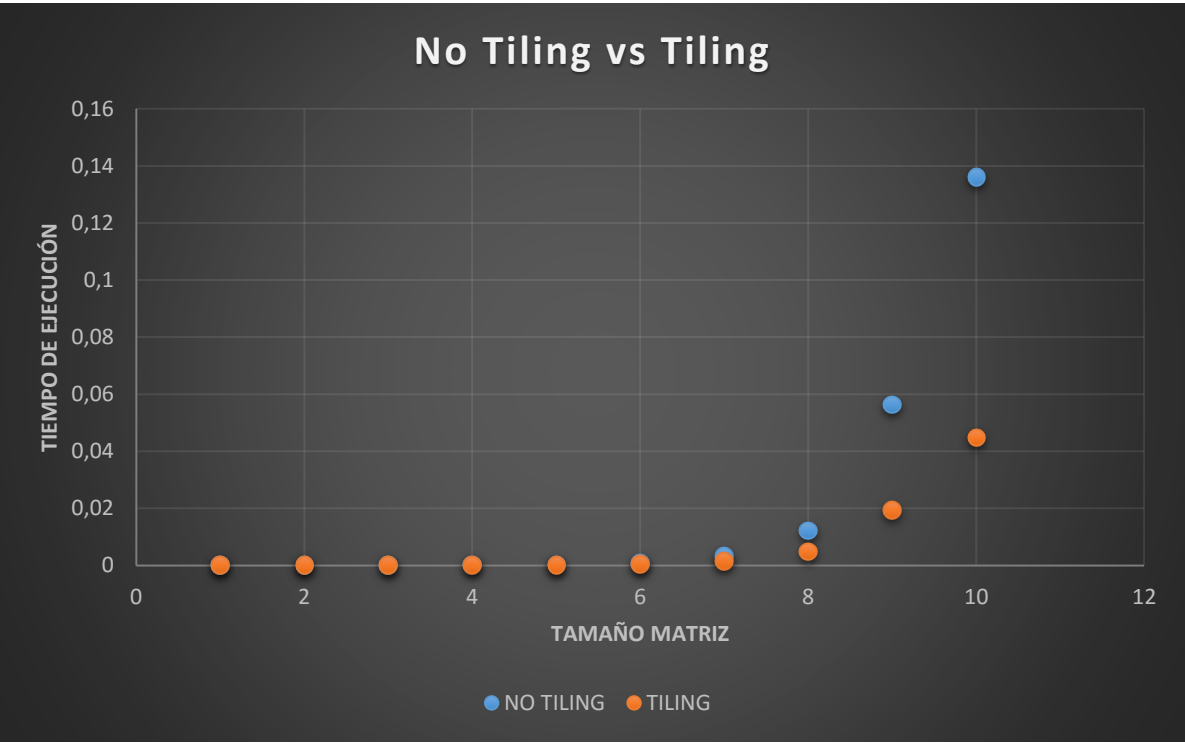


5. Graficas de tiempo de cada una de las implementaciones, Secuencial, Paralela y Paralela con Tile

Secuencial



Paralela y Paralela con Tiling



## 6. Tiempos de ejecución para diferentes tamaños de bloque (4x4, 16x16, 32x32)

BlockSize = 4

TILE WIDTH = 4

Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	320x256	256x240	0,051413	0,001869	0,001336	27,5082932	38,48278443
2	512x360	360x600	0,331106	0,009487	0,006668	34,90102245	49,65596881
3	1024x512	512x960	1,590748	0,045702	0,032112	34,80696687	49,53749377
4	2048x1024	1024x1200	7,541759	0,222044	0,155568	33,96515555	48,47885812
5	2048x2048	2048x1500	30,702244	0,548895	0,38335	55,93463959	80,08932829

BlockSize = 16

TILE WIDTH = 16

Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	320x256	256x240	0,051381	0,000759	0,000423	67,69565217	121,4680851
2	512x360	360x600	0,309866	0,003408	0,001718	90,92312207	180,3643772
3	1024x512	512x960	1,601256	0,012569	0,005511	127,3972472	290,5563419
4	2048x1024	1024x1200	9,528167	0,057041	0,022605	167,0406725	421,507056
5	2048x2048	2048x1500	29,93018	0,145155	0,055507	206,1946195	539,2145135

BlockSize = 32 Tile_Width = 32			Time				
Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	320x256	256x240	0,052593	0,000808	0,000424	65,09034653	124,0400943
2	512x360	360x600	0,329217	0,003337	0,001618	98,65657776	203,4715698
3	1024x512	512x960	1,635663	0,01231	0,004929	132,8727051	331,8447961
4	2048x1024	1024x1200	9,693118	0,056372	0,019458	171,9491592	498,1559256
15	2048x2048	2048x1500	30,109003	0,136082	0,044806	221,2563234	671,9859617

BlockSize = 32

TILE WIDTH = 4

A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
320x256	256x240	0,052374	0,000828	0,001509	63,25362319	34,70775348
512x360	360x600	0,318346	0,003424	0,001728	92,97488318	184,2280093
1024x512	512x960	1,68366	0,012117	0,002277	138,9502352	739,4202899
2048x1024	1024x1200	9,622837	0,055578	0,003173	173,141117	3032,725181
2048x2048	2048x1500	30,088345	0,135873	0,005348	221,4446211	5626,092932

**BlockSize = 4**

**TILE WIDTH = 32**

A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
320x256	256x240	0,052673	0,001881	0,000886	28,00265816	59,4503386
512x360	360x600	0,316848	0,009545	0,004439	33,19518072	71,37823834
1024x512	512x960	1,657431	0,045734	0,016981	36,24067433	97,60502915
2048x1024	1024x1200	9,59354	0,221967	0,079121	43,22056882	121,2515009
2048x2048	2048x1500	34,035694	0,548616	0,193856	62,03919317	175,5720432

## 7. Aceleración

La siguiente tabla indica el comportamiento de la aceleración variando el BlockSize y el TILE\_WIDTH para matrices de diferentes tamaños

Matrix Size	Acceleration Time Parallel				
	TILING BS4 - TW4	TILING BS16 - TW16	TILING BS32 - TW32	TILING BS32 - TW4	TILING BS4 - TW32
1	27,5082932	67,69565217	65,09034653	63,2536232	28,0026582
2	34,90102245	90,92312207	98,65657776	92,9748832	33,1951807
3	34,80696687	127,3972472	132,8727051	138,950235	36,2406743
4	33,96515555	167,0406725	171,9491592	173,141117	43,2205688
5	55,93463959	206,1946195	221,2563234	221,444621	62,0391932

Grafica sin el uso de Tiles

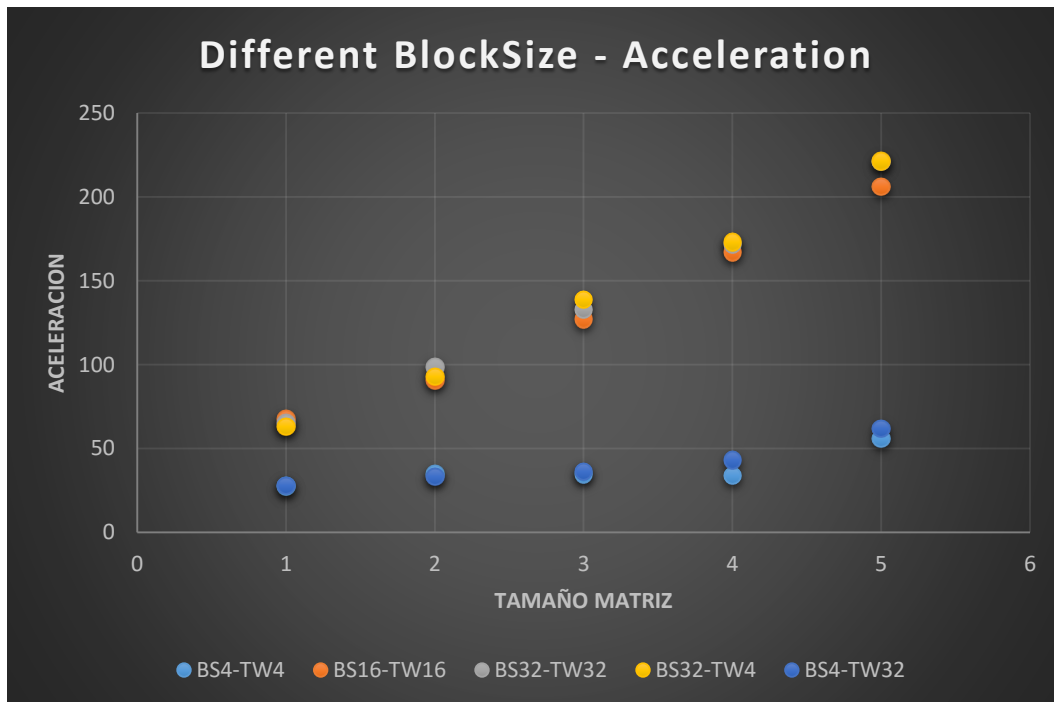
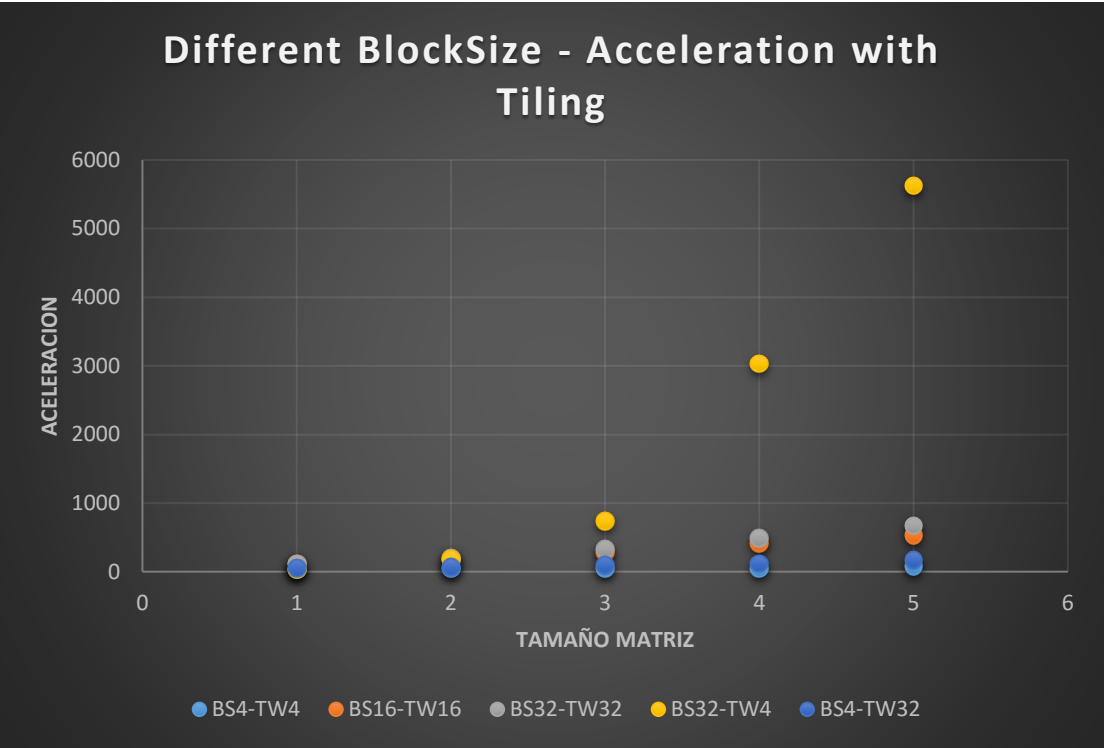


Tabla de aceleración con el uso de Tiles

	Acceleration Time Parallel with Tiling				
Matrix Size	TILING BS4 - TW4	TILING BS16 - TW16	TILING BS32 - TW32	TILING BS32 - TW4	TILING BS4 - TW32
1	38,4827844	121,468085	124,040094	34,7077535	59,4503386
2	49,6559688	180,364377	203,47157	184,228009	71,3782383
3	49,5374938	290,556342	331,844796	739,42029	97,6050292
4	48,4788581	421,507056	498,155926	3032,72518	121,251501
5	80,0893283	539,214513	671,985962	5626,09293	175,572043

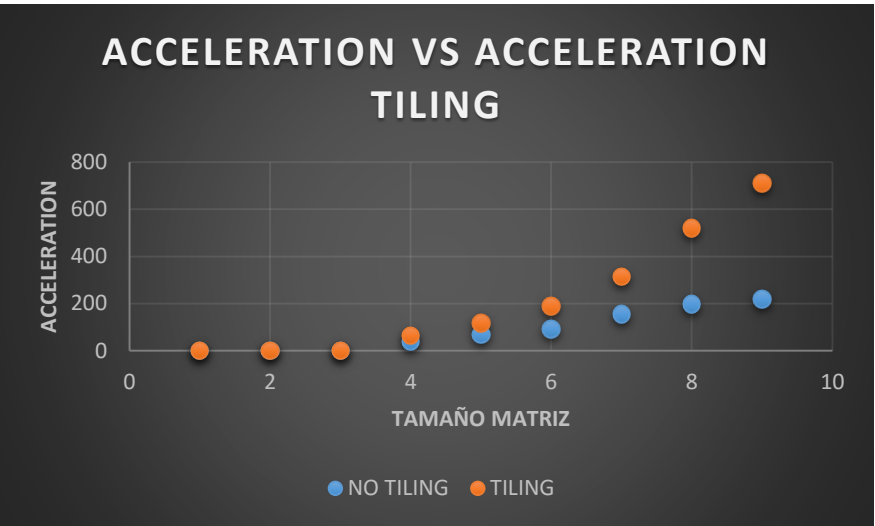


### 8. Valores para punto flotante

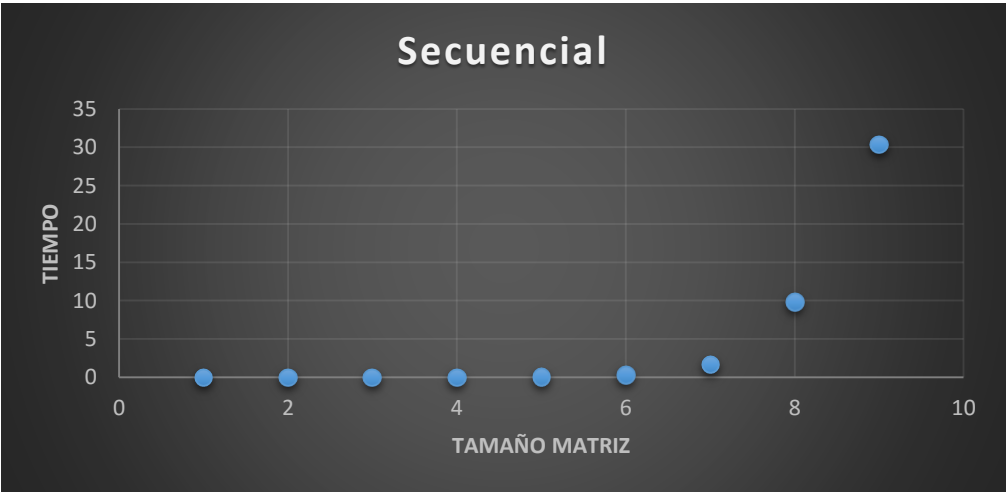
Tabla de datos para un BlockSize de 32 y un TILE\_WIDTH de 32

BlockSize = 32			Time				
Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	12x16	16x32	0,000018	0,000071	0,000042	0,253521127	0,428571429
2	32x64	64x10	0,000055	0,000091	0,000047	0,604395604	1,170212766
3	48x30	30x18	0,00007	0,000082	0,000046	0,853658537	1,52173913
4	128x128	128x256	0,012024	0,000303	0,000187	39,68316832	64,29946524
5	320x256	256x240	0,051744	0,000769	0,000441	67,28738622	117,3333333
6	512x360	360x600	0,317923	0,00346	0,001666	91,88526012	190,8301321
7	1024x512	512x960	1,665106	0,010659	0,005292	156,2159677	314,6458806
8	2048x1024	1024x1200	9,816488	0,049945	0,018823	196,5459606	521,5155926
9	2048x2048	2048x1500	30,34284	0,138691	0,042671	218,780166	711,0880926

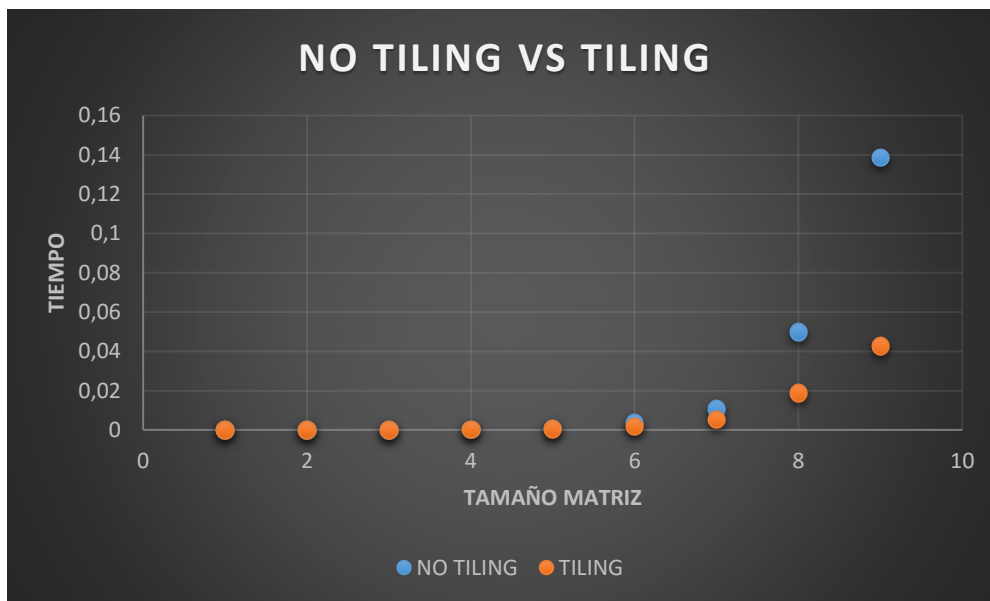
Grafica de aceleración y aceleración con el uso de Tiles



Graficas que ilustran el tiempo de ejecución para las tres formas, secuencial, paralela y paralela con Tiling







Tiempos de ejecución para diferentes tamaños de BlockSize y TILE\_WIDTH

**BlockSize = 4**  
**TILE WIDTH = 4**

Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	320x256	256x240	0,051742	0,001791	0,001338	28,89000558	38,67115097
2	512x360	360x600	0,318276	0,008998	0,006706	35,37186041	47,46137787
3	1024x512	512x960	1,642192	0,043922	0,032306	37,38882565	50,83241503
4	2048x1024	1024x1200	7,563054	0,21207	0,155701	35,6630075	48,57421597
5	2048x2048	2048x1500	34,383022	0,523941	0,382551	65,62384314	89,87826983

**BlockSize = 16**  
**TILE WIDTH = 16**

Matrix Size	A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
1	320x256	256x240	0,051598	0,000804	0,000427	64,17661692	120,8384075
2	512x360	360x600	0,315727	0,003654	0,001711	86,4058566	184,5277615
3	1024x512	512x960	1,640701	0,013176	0,00585	124,5219338	280,4617094
4	2048x1024	1024x1200	9,721459	0,05999	0,0223	162,0513252	435,9398655
5	2048x2048	2048x1500	30,241224	0,160597	0,054807	188,3050368	551,7766709

**BlockSize = 32**  
**TILE WIDTH = 4**

A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
320x256	256x240	0,051773	0,000788	0,001545	65,70177665	33,51003236
512x360	360x600	0,315907	0,003402	0,001754	92,85920047	180,1066135
1024x512	512x960	1,693665	0,010521	0,001927	160,9794696	878,9128179
2048x1024	1024x1200	9,732983	0,049439	0,003396	196,8685248	2866,01384
2048x2048	2048x1500	30,730125	0,139163	0,005573	220,8210875	5514,1082

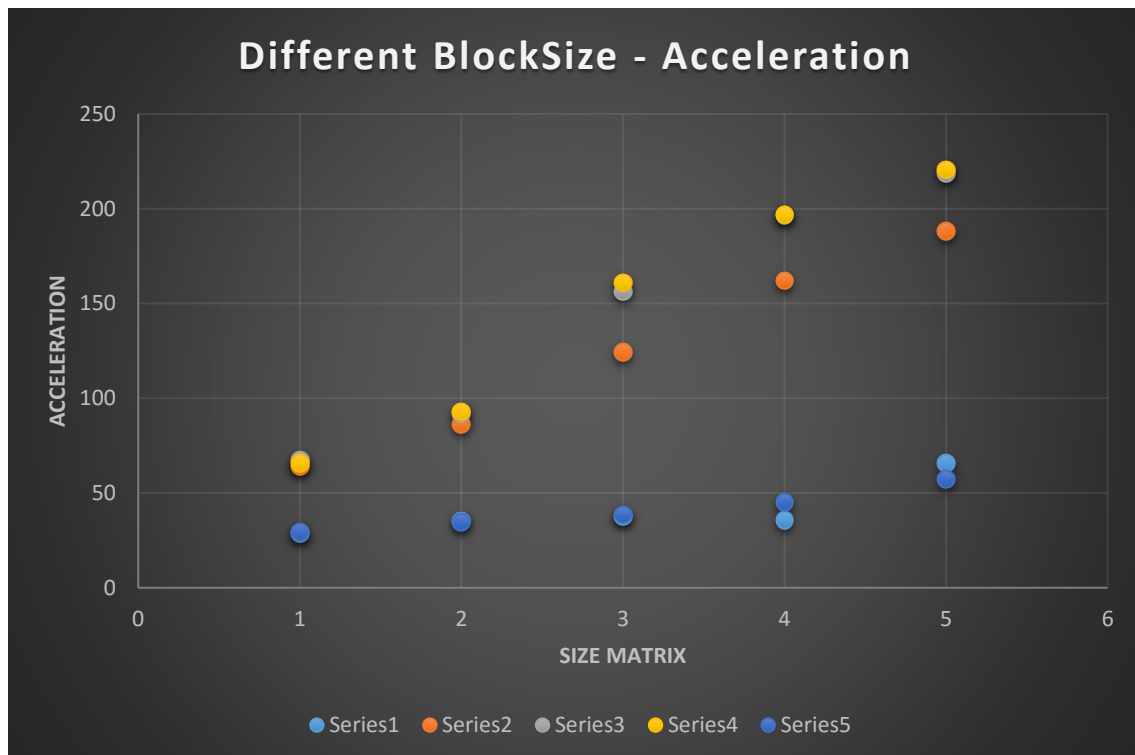
BlockSize = 4  
TILE WIDTH =  
32

A	B	SEQUENTIAL	NO TILING	TILING	Acceleration	Acceleration Tiling
320x256	256x240	0,051593	0,001763	0,000882	29,26432218	58,49546485
512x360	360x600	0,315598	0,009077	0,004494	34,76897653	70,22652425
1024x512	512x960	1,693433	0,043914	0,017016	38,56248577	99,52003996
2048x1024	1024x1200	9,606588	0,212889	0,078342	45,12486789	122,6237267
2048x2048	2048x1500	30,070133	0,524443	0,192569	57,33727593	156,1525116

Tabla de aceleración para los diferentes tamaños de BlockSize y TILE\_WIDTH

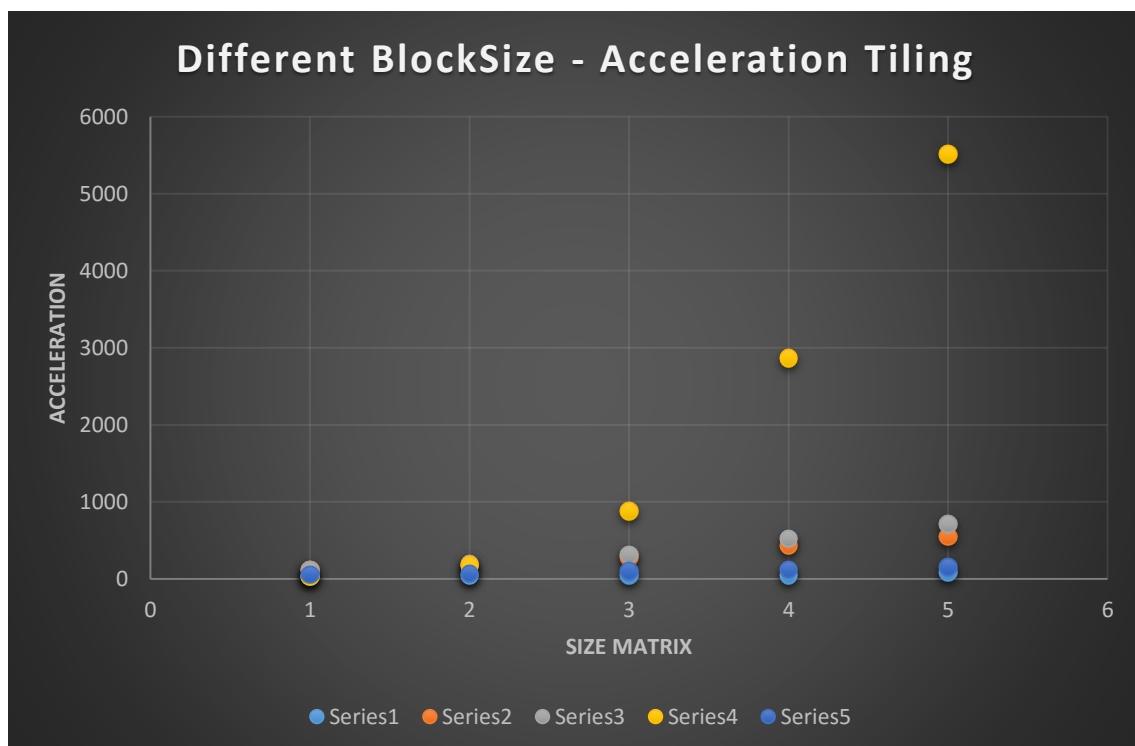
Matrix Size	Acceleration Time Parallel				
	NO_TILING BS4 - TW4	NO_TILING BS16 - TW16	NO_TILING BS32 - TW32	NO_TILING BS32 - TW4	NO_TILING BS4 - TW32
1	28,8900056	64,17661692	67,28738622	65,70177665	29,2643222
2	35,3718604	86,4058566	91,88526012	92,85920047	34,7689765
3	37,3888256	124,5219338	156,2159677	160,9794696	38,5624858
4	35,6630075	162,0513252	196,5459606	196,8685248	45,1248679
5	65,6238431	188,3050368	218,780166	220,8210875	57,3372759

Grafica correspondiente a la aceleración sin el uso de Tiles



Matrix Size	Acceleration Time Parallel with Tiling				
	TILING BS4 - TW4	TILING BS16 - TW16	TILING BS32 - TW32	TILING BS32 - TW4	TILING BS4 - TW32
1	38,67115097	120,8384075	117,3333333	33,5100324	58,4954649
2	47,46137787	184,5277615	190,8301321	180,106613	70,2265243
3	50,83241503	280,4617094	314,6458806	878,912818	99,52004
4	48,57421597	435,9398655	521,5155926	2866,01384	122,623727
5	89,87826983	551,7766709	711,0880926	5514,1082	156,152512

Grafica correspondiente a la tabla de aceleración implementando Tiling



## 9. Conclusiones

A partir de los valores tomados en cada implementación y observando el comportamiento de las gráficas se puede concluir:

- Se pueden presentar algunas alteración en los resultados ya que la herramienta utilizada es compartida
- Para matrices MXN podemos establecer que el tamaño del dimGrid está determinado es por la matriz resultante ya que esta es la que nos indica los bloques y los hilos que debemos de implementar, la diferencia con matrices MXM es notoria.
- En el caso de la matriz secuencial su tiempo de ejecución es óptimo cuando son valores pequeños es decir matrices inferiores a tamaño de 100, debido a que se toma un tiempo de pasar los datos del host al device, sin embargo para matrices de tamaño superior la diferencia de velocidad en el cómputo de los datos es notoria.

- Se destaca el comportamiento y el beneficio de usar memoria compartida, la diferencia al usar el algoritmo implementando Tiles con los demás algoritmos secuencial y paralelo normal es de destacarse reduciendo los tiempos de ejecución para matrices grandes.
- La aceleración observada en los dos casos que el algoritmo es implementado paralelamente es mucho mayor utilizando Tiles.
- Al realizar cambios en el tamaño del BlockSize y el TILE\_WIDTH se evidencia lo siguiente:
  - A medida que se aumenta el tamaño para el BlockSize y el Tile el funcionamiento es mucho mejor, optimizando los tiempos de ejecución, siempre y cuando el valor del BlockSize fuera igual al Tile.
  - El tiempo de ejecución que toma el algoritmo cuando el tamaño del BlockSize es el mayor (32) y el tamaño del Tile es igual a 4, se puede notar que el tiempo de ejecución para el algoritmo con Tiling es mucho mejor que los demás, donde su aceleración es descomunal en comparación a los diferentes tamaños.
- En cuanto a la implementación con números de tipo entero y números punto flotante, la diferencia en tiempos de ejecución no es muy notoria
- Las aceleraciones obtenidas con los algoritmos paralelos tienden a ser exponenciales esto a su vez se aumenta el tamaño de la matriz y no se sobrepase el límite de la tarjeta que brinda la plataforma o se esté usando.