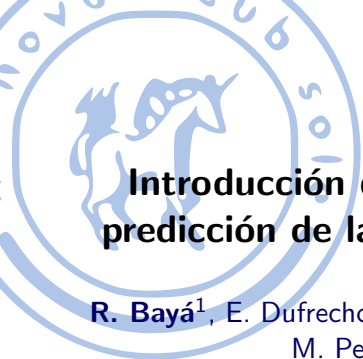




Introducción de técnicas de HPC para la predicción de la generación de energía solar

R. Bayá¹, E. Dufrechou¹, C. Porrini², A. Gutierrez², G. Cazes²,
M. Pedemonte¹ and P. Ezzatti¹

¹Instituto de la Computación, Universidad de la República (Montevideo, Uruguay).

²Instituto de Mecánica de los Fluidos e Ingeniería Ambiental, Universidad de la República (Montevideo, Uruguay).

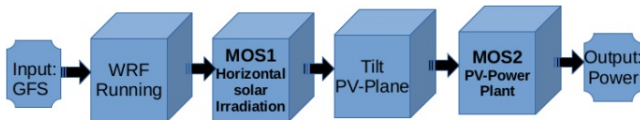
PRONOS 2016

Incorporación de Energías renovables en Uruguay.



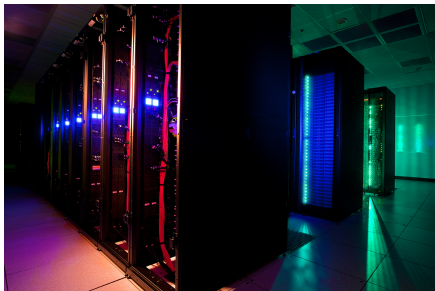
Necesario predecir la generación de energía para administrar de forma correcta la matriz energética.

En la Facultad de Ingeniería se desarrollaron herramientas para predecir la generación de energía eólica y solar.



Problema: Requiere un alto costo computacional, en particular el WRF.

Computación de alta performance (HPC).



Técnicas que permiten aprovechar diversos recursos de hardware.

Utilizar lo mejor posible la potencia de cómputo de una plataforma, haciendo un uso eficiente de sus componentes.

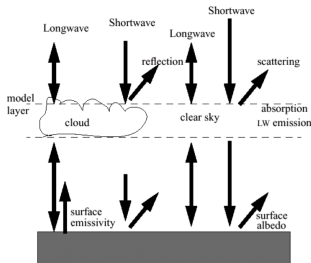
- 1 Propuesta
- 2 Evaluación experimental
 - Paralelismo de datos
 - Paralelismo masivo
- 3 Conclusiones

- 1 Propuesta
- 2 Evaluación experimental
 - Paralelismo de datos
 - Paralelismo masivo
- 3 Conclusiones

Propuesta

WRF - Modelo de predicción atmosférica.

Entre otras propiedades físicas y químicas simula la radiación emitida.



Radiación en forma de Onda corta y Onda larga.



Propuesta

Cuando se quiere predecir la radiación solar $\Rightarrow$ surgen varias implementaciones de radiación.

Difieren en:

- Precisión numérica.
- Tiempo de cómputo.

Todas implican un alto costo computacional, por encima de **1/3** del tiempo total!!.



Propuesta

Es deseable utilizar técnicas de alto desempeño para mejorar los tiempos de ejecución.

2 caminos:

- Estudiar el paralelismo de datos presente.
- Aplicar nuevas y distintas técnicas de HPC.



Propuesta

Analizar el paralelismo de datos y así ajustar la herramienta.

Método:

- Variar la cantidad de hilos que utiliza.
- Variar los modelos físicos utilizados.

Evaluar:

- Desempeño del problema.
- Escalabilidad del problema.



Propuesta

Introducir nuevo hardware de cómputo, en particular GPUs.

Método:

- Migrar distintas rutinas a esta plataforma.
- Priorizar rutinas altamente demandantes.

- 1 Propuesta
- 2 Evaluación experimental
 - Paralelismo de datos
 - Paralelismo masivo
- 3 Conclusiones

Evaluación experimental



Plataforma:

- Intel Xeon E5-2620, 6 cores at 2.0GHz $\times$ 2
- 128 GB of RAM
- NVIDIA GTX 480 (480 CUDA cores at 700 MHz and 1.5 GB of RAM)

Evaluación experimental



Se realizaron 3 casos de prueba.

Todos utilizan el modelo CAM2 para calcular la radiación.

Se varía entre casos de prueba:

- Módulo de microfísica.
- Módulo de capa de superficie.
- Módulo de capa limite.

Evaluación experimental



Tiempos de ejecución (en segundos) de los distintos casos de pruebas, variando la cantidad de hilos.

Esquema	# Hilos				
	4	8	10	12	14
PS1	6208	4148	3685	3514	3875
PS2	3597	2178	1869	1796	2058
PS3	6141	3489	2966	2718	3187

Evaluación experimental



Analizando los datos se observa:

- PS2 es la versión que presenta mejor desempeño - $2.3\times$
- PS3 presenta un buen desempeño - $2.0\times$
- PS3 ofrece mejor escalabilidad que PS2.
- Ninguna versión presenta mejora cuando se utilizan más de 12 hilos.

Evaluación experimental



Se utilizan variantes de 3 rutinas en GPU:

- bdy_interp1 (Silva et al.)
- WSM5 (Michalakes et al.)
- Scalar_Advection (Michalakes et al.)

Con la nueva rutina bdy_interp1 en GPU se alcanzó un speedup de 33x para esta función.

Agregando WSM5 y Scalar_Advection se consiguió una mejora del 30% en los tiempos totales del modelo.

- 1 Propuesta
- 2 Evaluación experimental
 - Paralelismo de datos
 - Paralelismo masivo
- 3 Conclusiones

Conclusiones



Se realizó un estudio de la aplicación WRF variando:

- Modelos físicos.
- Cantidad de hilos.

Con estos resultados, se estudió el desempeño y la escalabilidad de la aplicación, considerando además la precisión del modelo.

Se obtuvieron 2 configuraciones distintas, que resultan atractivas en diferentes escenarios.

Conclusiones



Se introdujeron mejoras que utilizan la GPU como unidad de cómputo sobre 3 funciones:

- bdy_interp1
- WSM5
- Scalar_Advection

Utilizando estas mejoras, se logró que los tiempos de cómputo del WRF fueran un **30%** más eficientes.



GRACIAS.

ICT4V

Information and Communication
Technologies for Verticals

