
	Propuesta trabajo de grado Departamento EEST	<b>Código</b>	00
		<b>Página</b>	1 de 10

## **COMPUTACIÓN PARALELA EN PYTHON SOBRE UN CLUSTER DE ALTO RENDIMIENTO**

**Carlos Antonio Galán Guerra**  
Estudiante Ing. de Sistemas


**José Orlando Maldonado Bautista**  
Director de tesis

**DEPARTAMENTO DE INGENIERÍAS ELÉCTRICA ELECTRÓNICA SISTEMAS Y  
TELECOMUNICACIONES  
FACULTAD DE INGENIERÍAS Y ARQUITECTURA  
INGENIERIA DE SISTEMAS  
UNIVERSIDAD DE PAMPLONA  
PAMPLONA  
JUNIO de 2016**

	<p style="text-align: center;">Propuesta trabajo de grado Departamento EEST</p>	<b>Código</b>	00
		<b>Página</b>	1 de 10

### 1. INFORMACIÓN GENERAL DEL TRABAJO DE GRADO

<b>Título: COMPUTACION PARALELA EN PYTHON SOBRE UN CLUSTER DE ALTO RENDIMIENTO</b>			
Nombre Autor: <b>Carlos Antonio Galán Guerra</b>		C.C. : <b>1094268230</b>	
E-mail:		FIRMA:	
Lugar de Ejecución del Proyecto: Pamplona, Norte de Santander		Teléfono: <b>3016548484</b>	
Duración de Proyecto (en semanas): <b>16</b>			
<b>Modalidad:</b>			
Investigación	<input checked="" type="checkbox"/>	Práctica Empresarial*	<input type="checkbox"/>
Diplomado	<input type="checkbox"/>	Docencia	<input type="checkbox"/>
<i>*Deberá adjuntar carta de aceptación de la empresa o copia del convenio o contrato</i>			
<b>Objetivo General</b>			
<ul style="list-style-type: none"> <li>• Diseño de un clúster de alto rendimiento para la resolución de problemas complejos usando computación paralela con Python</li> </ul>			
<b>Objetivos específicos</b>			
<ul style="list-style-type: none"> <li>• Estudio de potencialidades que ofrece Python en la computación de alto rendimiento</li> <li>• Desarrollo de un clúster de alto rendimiento usando el módulo "Parallel Python"</li> <li>• Evaluación del desempeño de la plataforma computacional, a partir de un estudio comparativo de la solución de un problema complejo</li> </ul>			
Valor Total del Proyecto:			
<b>Aceptación director de la tesis:</b>		<b>Autorización director del programa: SI NO</b>	
NOMBRE: <b>José Orlando Maldonado Bautista</b>		FIRMA:	
FIRMA:			
Jurado 1:		Firma Jurado 1:	
Jurado 2:		Firma Jurado 2:	
Jurado 3:		Firma Jurado 3:	
<b>Sustentación Anteproyecto:</b>			
Lugar : _____ Fecha: _____ Hora _____			
Aprobado		Rechazado	
Incompleto			
<b>Observaciones del jurado:</b> _____			
_____			
_____			
_____			
Firma Jurado 1: _____ Firma Jurado 2: _____ Firma Jurado 3: _____			
<b>Realización de Correcciones</b>			
Lugar : _____ Fecha: _____ Hora _____			

	<p style="text-align: center;">Propuesta trabajo de grado Departamento EEST</p>	<b>Código</b>	00
		<b>Página</b>	1 de 10

<b>Aprobado</b>	<b>Rechazado</b>
Firma Jurado 1: _____	Firma Jurado 2: _____
	Firma Jurado 3: _____

### RESUMEN DEL PROYECTO

Los Clúster han aparecido como una herramienta computacional de alto rendimiento para la solución de problemas en ciencias e ingeniería. Es así, como en la siguiente propuesta de investigación nosotros proponemos estudiar las potencialidades que ofrece Python en computación de alto rendimiento y así mismo, la implementación de una plataforma Clúster de alto rendimiento usando un módulo “Parallel Python”. La evaluación del desempeño de la plataforma propuesta se hará haciendo un estudio comparativo de problemas que requieren altos tiempos de duración de procesamiento de información y que a su vez son de interés del grupo de investigación CICOM (Grupo de Ciencias Computacionales) y otros de la Universidad de Pamplona.

**Palabras claves:** Clúster, Parallel Python, Computación de alto rendimiento


### PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA.

En la actualidad es conocido que los equipos de cómputo han aumentado su capacidad de procesamiento considerablemente, sin embargo también es cierto que la solución de muchos problemas requiere en ocasiones un consumo mucho mayor en términos computacionales. En ese sentido, para la solución de un problema es muy importante tener en cuenta los tiempos que lleva el proceso o tarea de computo, debido a que este puede limitar la capacidad de tomar decisiones oportunas. Es por esto que nosotros pensamos que una alternativa a las limitaciones antes expuestas, son los clúster, en nuestro caso proponemos un clúster de alto rendimiento que permita dividir el procesamiento en distintos equipos de cómputo conectados a una red, donde la única limitante será la cantidad de equipos que podamos agregar a la red. Así nosotros esperamos que dicha configuración nos permita optimizar los tiempos de duración del procesamiento de datos o información y a su vez también permitirá abordar problemas que necesiten una capacidad de procesamiento mayor, en tiempos accesibles y de bajo costo.

### MARCO TEÓRICO

#### **Computación de alto rendimiento (HPC):**

La computación de alto rendimiento es un concepto importante a la hora de abordar


	<p>Propuesta trabajo de grado</p> <p>Departamento EEST</p>	<b>Código</b>	00
		<b>Página</b>	1 de 10

soluciones a problemas complejos que consumen recursos como tiempo, entre otros. Para lograr cumplir esta finalidad la computación de alto rendimiento se apoya en distintas tecnologías computacionales como lo son las supercomputadoras, clúster o también la computación paralela. La mayoría de las ideas actuales de la computación distribuida se han basado en la computación de alto rendimiento.

## **Paradigmas y herramientas para HPC**

### **Modelos Paralelos Estándares: Paradigmas**

Aquellos modelos que describen estructuras o patrones típicos para resolver un problema, se los denomina paradigmas. Un paradigma, muy conocido, es el paradigma para la resolución de problemas divide y vencerás. Este se caracteriza por resolver un problema a través de la solución de sub-problemas más pequeños y de la misma naturaleza que el problema original; en paralelismo, existen varios paradigmas. El origen de ellos se debe a la naturaleza de los sistemas paralelos y a los distintos aspectos que se pueden paralelizar en un problema. La idea de dar una solución paralela se basa en la filosofía: *un problema grande es dividido en sub-problemas cada uno de los cuales se resuelve en forma concurrente al resto* <sup>[1]</sup>. Cuando se construye algoritmos paralelos se debe considerar distintos aspectos: 1) La naturaleza del problema, 2) La relación entre los datos, 3) La posibilidad de dividir los datos o el problema y 4) La independencia o dependencia de los resultados. Todo esto hace que surjan distintos paradigmas de computación paralela. Es así como una clasificación se deriva de considerar el objeto a paralelizar: los datos o la computación. Dos categorías surgen de esta clasificación el Paralelismo de Datos y el Paralelismo de Control <sup>[2]</sup>. En Foster <sup>[3]</sup>, si la técnica de paralelismo considera primero los datos asociados al problema, luego determina una apropiada partición y finalmente trabaja sobre ellos, se la denomina Paralelismo por descomposición de dominios. En cambio cuando se prioriza la descomposición de la computación sobre los datos lo denomina paralelismo por descomposición funcional. Otros paradigmas paralelos son aquellos orientados a la arquitectura. Los más conocidos son aquellos que consideran la memoria de la máquina paralela como un recurso compartido, paradigma de memoria compartida o paradigma de memoria distribuida <sup>[5]</sup>.

	<p>Propuesta trabajo de grado</p> <p>Departamento EEST</p>	<b>Código</b>	00
		<b>Página</b>	1 de 10

### **Paradigmas Orientados a la Arquitectura:**


Una computadora convencional consiste de un procesador, el cual ejecuta un programa almacenado en su memoria. El procesador puede acceder a cada dirección de memoria. Extender el modelo de único procesador, significa tener múltiples procesadores y múltiples módulos de memorias conectados de alguna manera. La topología de la conexión entre los procesadores y los módulos de memoria está determinada por cómo se considera a la memoria: única o propia. En el primer caso todos los módulos integran un único espacio de direcciones; para el segundo caso, la memoria se considera propia de cada procesador <sup>[5]</sup>.

### **CLUSTER DE ALTO RENDIMIENTO**

El objetivo de un clúster de alto rendimiento es compartir el recurso más valioso de un computador, es decir, la capacidad de procesamiento. Este tipo de clústeres son muy populares, y quizás también los más antiguos, ya que en sus inicios fueron desarrollados para centros de cómputo enfocados en investigaciones científicas donde era necesario tener un gran poder de procesamiento, capaz de soportar una inmensa cantidad de cálculos para resolver un problema específico<sup>[8]</sup>.

### **PYTHON**

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multi-paradigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multi-plataforma. Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública

	<p>Propuesta trabajo de grado</p> <p>Departamento EEST</p>	<b>Código</b>	00
		<b>Página</b>	1 de 10

general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores. [6]

### **Características y paradigmas:**

Python es un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones. Python usa tipado dinámico y conteo de referencias para la administración de memoria. Una característica importante de Python es la resolución dinámica de nombres; es decir, lo que enlaza un método y un nombre de variable durante la ejecución del programa (también llamado enlace dinámico de métodos). Otro objetivo del diseño del lenguaje es la facilidad de extensión. Se pueden escribir nuevos módulos fácilmente en C o C++. Python puede incluirse en aplicaciones que necesitan una interfaz programable. Aunque la programación en Python podría considerarse en algunas situaciones hostiles a la programación funcional tradicional del Lisp, existen bastantes analogías entre Python y los lenguajes minimalistas de la familia Lisp como puede ser Scheme. [6]


### **LIBRERIAS PARA COMPUTACION EN PARELO.**

#### **Parallel Python (PP):**

Parallel Python es un módulo de Python que proporciona el mecanismo para la ejecución en paralelo de código Python en SMP (Sistemas con Múltiples Procesadores) y Clúster (Ordenadores conectados vía red), es ligero, fácil para la instalación y tiene la posibilidad de integrarse con otro software creado en Python. Hoy en da el software escrito en Python encuentra una amplia gama de categorías incluyendo los negocios, el análisis de datos y cálculos científicos. Esto junto con la amplia disponibilidad de ordenadores SMP y de Clúster en el mercado crea la demanda para la ejecución paralela de código Python. [7]

#### ➤ **Características**

- Ejecución paralela de Python en SMP (Sistema de Multiprocesamiento) y Clúster.
- Fácil de entender y poner en práctica la técnica.

	<p>Propuesta trabajo de grado</p> <p>Departamento EEST</p>	<b>Código</b>	00
		<b>Página</b>	1 de 10

- Detección automática de la configuración óptima.
- Asignación de procesos dinámicos (en número de procesos trabajando puede ser cambiado en tiempos de ejecución).
- Equilibrio de carga.
- Autenticación basada en SHA (secure Hash Algorithm, Algoritmo de hash seguro; es un conjunto de funciones hash diseñadas por la Agencia de Seguridad Nacional de los Estados Unidos) para conexiones de red.
- Fácil portabilidad para Windows, Linux, Unix.
- Open Source (Código abierto).

## **DELIMITACIÓN**

### **OBJETIVO GENERAL**


Diseño de un clúster de alto rendimiento para la resolución de problemas complejos usando computación paralela con Python.

### **OBJETIVOS ESPECÍFICOS**

- Estudio de potencialidades que ofrece Python en la computación de alto rendimiento.
- Desarrollo de un clúster de alto rendimiento usando el módulo “Parallel Python”.
- Evaluación del desempeño de la plataforma computacional, a partir de un estudio comparativo de la solución de un problema complejo.

### **ACOTACIONES :**

- Para la selección del problema complejo a solucionar mediante computación en paralelo, se tendrán en cuenta algunos problemas abordados por proyectos con los que está vinculado el grupo de investigación en Ciencias Computacionales, como por ejemplo el modelado numérico de la ecuación de onda en 3D, filtrado y análisis de fourier en imágenes tridimensionales, entre otras.

	Propuesta trabajo de grado Departamento EEST	<b>Código</b>	00
		<b>Página</b>	1 de 10

## TAREAS Y CRONOGRAMA DE ACTIVIDADES

Nosotros proyectamos desarrollar la investigación mediante una metodología compuesta de 6 fases fundamentales, la cual se espera ejecutar en un periodo de 16 semanas. Además de esto es importante mencionar, que este problema ha venido siendo abordado desde principio del presente año. A continuación presentamos resumidamente una descripción de las actividades propuestas en el cronograma de actividades (Ver tabla 6.1).

**a. Revisión del estado del arte.**

Estudio de los conceptos básicos de computación de alto rendimiento serán estudiados a lo largo tiempo de duración del proyecto, así mismo los diferentes aportes que propone la comunidad científica nacional y/o internacional.

**b. Programación paralela con Python.**

Se estudiará conceptos para la programación paralela con Python.

**c. Exploración de herramientas de desarrollo.**

Se estudiaran lenguajes y/o herramientas de desarrollo, lenguajes, IDEs.

**d. Estudio en profundidad de Parallel Python: Directivas, Clausulas, Parámetros. Ejemplos didácticos**

**e. Identificación de características de los problemas susceptibles de optimizar. (Selección de problemas de prueba)**


**f. Evaluación de plataforma**

Una vez seleccionado el problema de prueba se procede a la implementación y pruebas para evaluar la plataforma. Las pruebas estarán direccionadas a codificación y análisis de rendimiento.

**g. Análisis, presentación y socialización de resultados**

Los resultados obtenidos con la ejecución del proyecto serán presentados a través de un manuscrito, así mismo se presentaran y socializaran en eventos académicos-científicos organizados en la Universidad y a nivel nacional. Además la sustentación del trabajo para optar al título de Ing. de Sistemas será presentado al finalizar el proyecto.



	Propuesta trabajo de grado Departamento EEST	<b>Código</b>	00
		<b>Página</b>	1 de 10


ACTIVIDADES	TIEMPO (SEMANAS)																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
a	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
b	■	■	■	■														
c			■	■	■	■												
d					■	■	■	■										
e							■	■	■	■								
f									■	■	■	■	■					
g									■	■	■	■	■	■	■	■		

Tabla 6.1. Cronograma y descripción de Actividades

## 7 RESULTADOS/PRODUCTOS ESPERADOS Y POTENCIALES BENEFICIARIOS

### 7.1. Impactos esperados a partir del uso de los resultados:

Impactos	Plazo	Indicador	Supuestos
Sociales	corto	Socialización de resultados	Inicialmente los resultados obtenidos serán socializados en el semillero de investigación a fin dar soluciones a los problemas planteados en grupos afines a CICOM. Esperamos también que este trabajo genere un gran impacto dentro de los distintos grupos investigación de la Universidad de Pamplona, que requieren la solución de problemas específicos a sus estudios. Esperamos que el uso de la computación de alto rendimiento sea una puerta de entrada a la solución de problemas de frontera del conocimiento en distintas áreas del conocimiento.
	Mediano	Publicación de resultados	La documentación generada y el reporte técnico estará disponible para consulta en línea desde la web del programa, con el fin de facilitar la continuidad del trabajo. Así mismo el documento de tesis estará disponible en Biblioteca de la Universidad.

	Propuesta trabajo de grado Departamento EEST	<b>Código</b>	00
		<b>Página</b>	1 de 10

	Mediano	Desarrollo de proyectos de aplicación	Con la finalización de este Proyecto se espera identificar problemas de los distintos grupos de investigación de la Universidad, y así puedan ser abordados desde este paradigma de programación.
--	---------	---------------------------------------	---

Tabla 7.1. **Impactos esperados después de finalizar el proyecto**

## REFERENCIAS BIBLIOGRAFICAS

[1] BLELLOCH, Guy E. Programming parallel algorithms. *Communications of the ACM*, 1996, vol. 39, no 3, p. 85-97.

[2] CRANDALL, Phyllis; QUINN, Michael J. Block Data Decomposition for Data-Parallel Programming on a Heterogeneous Workstation Network. En *HPDC*. 1993. p. 42-49.

[3] FOSTER, Ian. Designing and building parallel programs. 1995.

[4] Computación de alto rendimiento, 2016. *Es.wikipedia.org* [online]

[5] M. F. Piccoli, *Computación de alto desempeño en GPU*. 2011.

[6] Python, 2016. *Es.wikipedia.org* [online]

[7] Parallel Python , 2016. *Parallelpython.com* [online]

[8] G. Cáceres, “Estrategia de Implementación de un Clúster de Alta Disponibilidad de N nodos sobre Linux usando Software Libre,” p. 172, 2012.