

## Contenido

<u>1. INTRODUCCIÓN</u> .....	1
1.1 ARQUITECTURA .....	2
1.1.1 Segmentación .....	2
1.1.2 Paralelismo .....	3
1.2 PROCESADORES VECTORIALES SEGMENTADOS .....	5
1.2.1 Arquitectura.. .....	5
1.2.2 Lenguajes de programación.....	7
1.2.3 Compiladores .....	9
1.3 SISTEMAS MULTIPROCESADOR.. .....	10
1.3.1 Arquitectura .....	10
1.3.2 Lenguajes de programación.....	13
1.3.3 Compiladores .....	14
1.4 PRESENTACIÓN DE LA TESIS Y APORTACIONES .....	15
<u>2. TÉCNICAS DE PARALELIZACIÓN AUTOMÁTICA</u> .....	17
2.1 PROCESO DE PARALELIZACIÓN .....	18
2.2 RELACIONES DE DEPENDENCIA ENTRE SENTENCIAS.....	19
2.2.1 Dependencias de datos.....	20
2.2.2 Dependencias de control.....	22
2.2.3 Grafode dependencias.....	23
2.2.4 Detección de dependencias .....	25
2.2.5 Notación y definiciones .....	26
2.3. TÉCNICAS DE PARALELIZACIÓN .....	27
2.3.1 Distribución de bucles .....	28
2.3.2 Intercambio de bucles .....	32
2.3.3 Eliminación de dependencias .. .....	33
2.3.3.1 Cambio de nombre y Expansión escalar .. .....	34
2.3.3.2 Refinamiento de Nodos .....	34
2.3.4.DOPIPE.....	35

2.3.5 DOACROSS .....	37
2.3.6 Particionado Parcial .....	40
2.3.7 Contracción de Ciclos .....	41
2.3.8 Reconocimiento de patrones .....	44
2A BUCLES CONDICIONALES .....	46
2.4.1 Vectorización de bucles condicionales .....	47
2.4.2 Paralelización de bucles condicionales .....	49
<u>3. EVALUACIÓN DEL PARALELISMO Y LONGITUD VECTORIAL ...</u>	<u>51</u>
3.1 PARALEUSMODEUNBUCLE .....	52
3.1.1 Bucles con GDA .....	52
3.1.2 Bucles con una única recurrencia hamiltoniana .....	54
3.1.3 Bucles con GDC .....	56
3.2 LONGITUD VECTORIAL DE SENTENCIAS EN BUCLES .....	58
3.2.1 Bucles con GD A .....	58
3.2.2 Bucles con GDC .....	59
3.3 INFLUENCIA DEL TIEMPO DE EJECUCIÓN .....	61
3.4 SENTENCIAS CONDICIONALES .....	62
3.4.1 Cotas del paralelismo .....	63
3.4.1.1 Cota superior del paralelismo .....	63
3.4.1.2 Cota inferior del paralelismo .....	68
3.4.2 Modelo probabilístico del paralelismo .....	73
3.5 EFICIENCIA DEL PROCESO DE PARALELIZACIÓN .....	81
<u>4. GRAPH TRAVERSE SCHEDULING. Paralelización de Recurrencias</u>	<u>84</u>
4.1 GRAPH TRAVERSE SCHEDULING .....	85
4.2 GRAFOS CON UNA ÚNICA RECURRENCIA HAMILTONIANA .....	85

4.3 GRAFOS HAMILTONIANOS CON VARIAS RECURRENCIAS .....	88
4.3.1 Distribución de operaciones .....	88
4.3.2 Sincronización .....	88
4.4 GENERACIÓN DE CÓDIGO PARALELO .....	94
4.4.1 Reescritura del bucle secuencial.....	94
4.4.2 Instrucciones de sincronización.....	97
4.5 TRANSFORMACIONES DEL GRAFO DE DEPENDENCIAS...	98
4.5.1 Obtención de una recurrencia $R_{sch}$ .....	99
4.5.1 Reducción del número de tareas. ....	100
4.5.1 Paralelismo sin sincronización y reducción del número de sincronizaciones .....	102
4.6 SENTENCIAS CONDICIONALES .....	104
4.6.1 Distribución de operaciones.....	104
4.6.2 Sincronización .....	106
<u>5. GRAPH TRAVERSE SCHEDULING. Vectorización deRecurrencias</u>	112
5.1 GRAFOS CON UNA ÚNICA RECURRENCIA HAMILTONIANA .....	113
5.2 GRAFOS HAMILTONIANOS CON VARIAS RECURRENCIAS .....	113
5.3 GENERACIÓN DE CÓDIGO VECTORIAL.....	117
5.4 MODIFICACIONES DEL GRAFO DE DEPENDENCIAS .....	119
5.4.1 Obtención de $R_{sch}$ .....	119
5.4.2 Condiciones de vectorización .....	121
5.4.3 Facilitar la generación de código vectorial.....	123

<u>6. EVALUACIÓN Y COMPARACIÓN DE MÉTODOS</u> .....	125
6.1 GENERACIÓN DE GRAFOS ALEATORIOS.....	126
6.2 VARIACIÓN DEL PARALELISMO Y TIEMPO DE EJECUCIÓN VECTORIAL.....	127
6.3 EVALUACIÓN DE LA GENERACIÓN DE CÓDIGO VECTORIAL.....	132
6.4 COMPARACIÓN DE MÉTODOS DE PARALELIZACIÓN DE RECURRENCIAS.....	135
<u>7. CONCLUSIONES Y LÍNEAS ABIERTAS</u> .....	142
7.1 CONCLUSIONES.....	143
7.2 LÍNEAS ABIERTAS.....	145
<i>REFERENCIAS</i> .....	146