GCM-Gross

$\vec{v}^0, p^0, T^0, S^0, \rho^0 \leftarrow$ *compute_initial_ocean_states*()

for each time step n

Perform the following computation

Step 1	$\vec{G_v}^*$	$\leftarrow$	*estimate_Gv*($\vec{v}^n$,p^n,ρ^n)
Step 2	p^{n+1}	$\leftarrow$	*diagnose_pressure*(p^n,$\vec{v}^n$,$\vec{G_v}^*$)
Step 3	$\vec{v}^{n+1}$	$\leftarrow$	*advance_velocity*(p^{n+1},v^n,$\vec{G_v}^*$)
Step 4	T^{n+1}	$\leftarrow$	*advance_temperature*(T^n,$\vec{v}^{n+1}$)
Step 5	S^{n+1}	$\leftarrow$	*advance_salinity*(S^n,$\vec{v}^{n+1}$)
Step 6	ρ^{n+1}	$\leftarrow$	*calculate_density*(T^{n+1},S^{n+1})

return($\vec{v}^{final}, p^{final}, T^{final}, S^{final}, \rho^{final}$)

τ **is the tolerance,** p^0 **is an initial estimate of** p^{final}

Elliptic_Solver(**p^0**,*Gdiv*,*Vdiv*)

$f \leftarrow$ *compute_source*(*Gdiv*,*Vdiv*)

$r^0 \leftarrow f - \mathbf{A}p^0$;

$b^0 \leftarrow \mathbf{K}r^0$

$\xi^0 \leftarrow b^0$

while $\sqrt{r^i \cdot r^i} > \tau$ **do**

$\gamma \leftarrow (r^i \cdot \xi^i)$

$\alpha \leftarrow \gamma \ / \ (b^i \cdot \mathbf{A}b^i)$

$p^{i+1} \leftarrow p^i + \alpha b^i$

$r^{i+1} \leftarrow r^i - \alpha \mathbf{A} b^i$

$\xi^{i+1} \leftarrow \mathbf{K}r^{i+1}$

$\beta \leftarrow (r^{i+1} \cdot \xi^{i+1}) \ / \ \gamma$

$b^{i+1} \leftarrow \xi^{i+1} + \beta b^i$

$i \leftarrow i + 1$

return(p^{final})

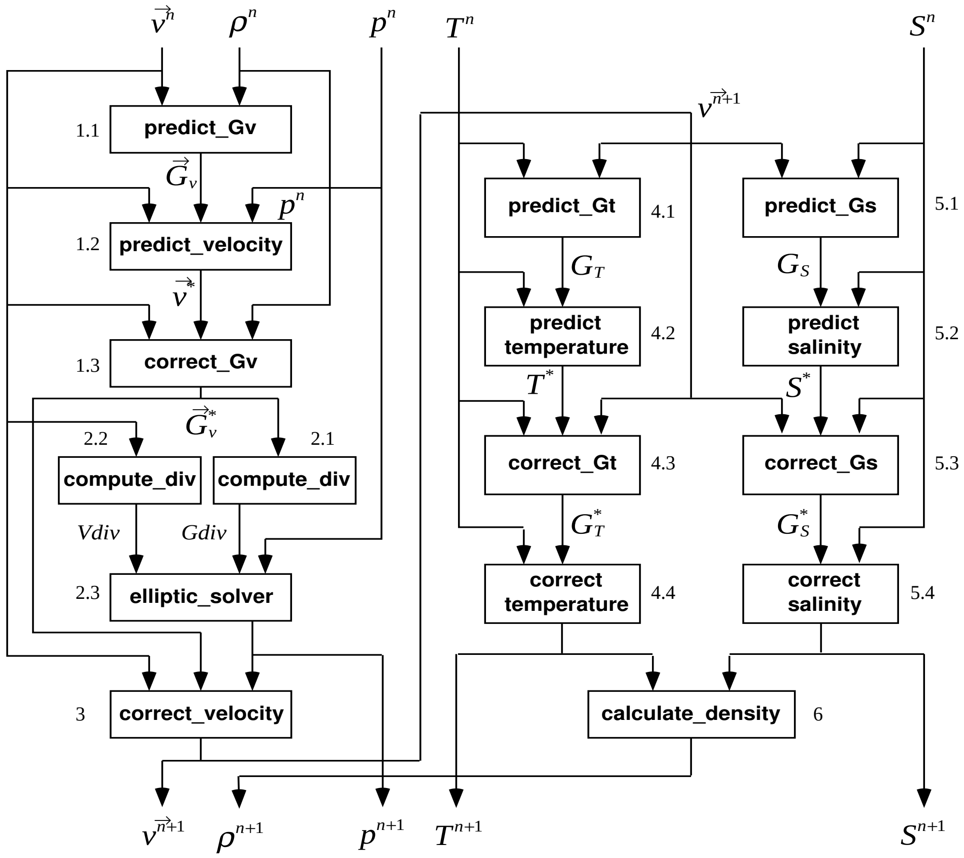
GCM **()**

$\vec{v}^0, p^0, T^0, S^0, \rho^0 \leftarrow$ *compute_initial_ocean_states()*

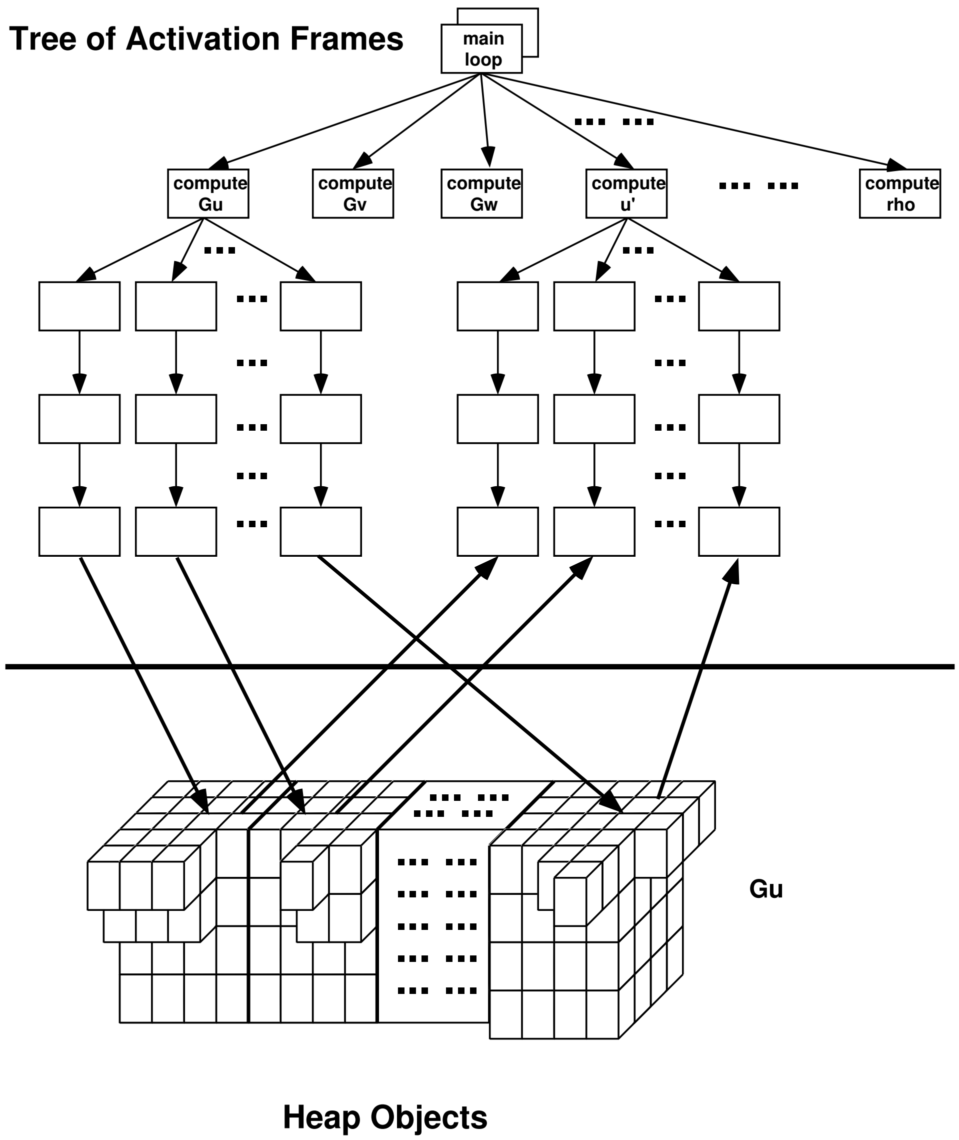
for each time step n

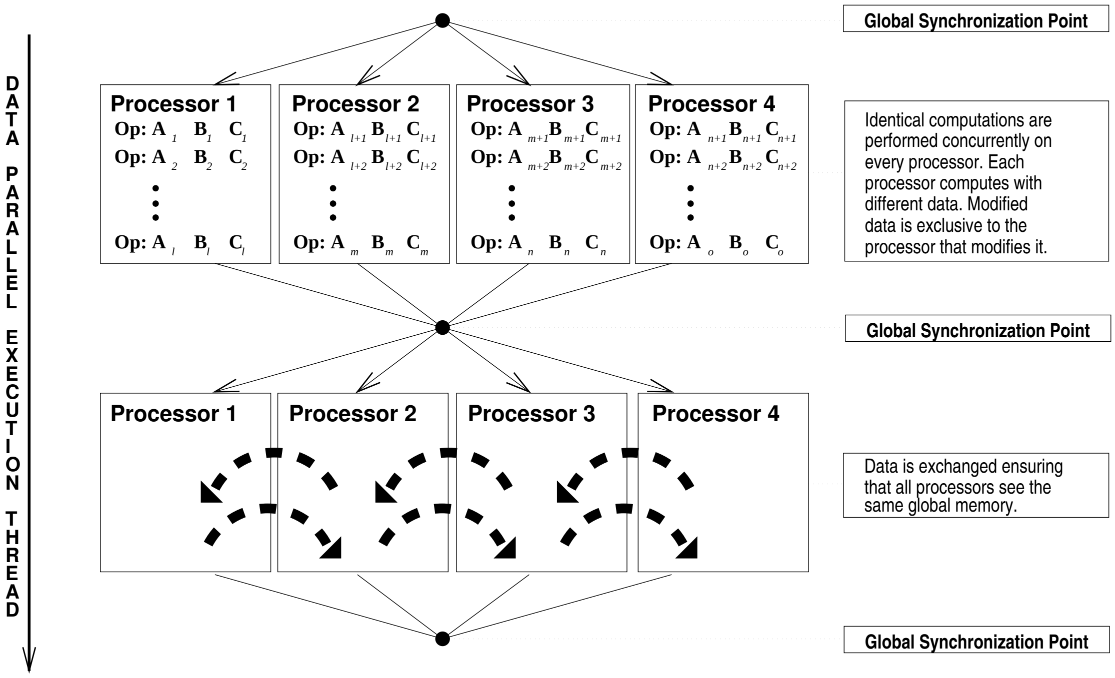
Step 1.1	$\vec{G}_v$	$\leftarrow$	*predict_Gv*($\vec{v}^n$,ρ^n)	**170 FPops**		
Step 1.2	$\vec{v}^*$	$\leftarrow$	*predict_velocity*($\vec{v}^n$,p^n,$\vec{G}_v$)	**20 FPops**		
Step 1.3	$\vec{G}_v^{\ *}$	$\leftarrow$	*correct_Gv*($\vec{v}^n$,$\vec{v}^*$,ρ^n)	**277 FPops**		
Step 2.1	*Gdiv*	$\leftarrow$	*compute_div*($\vec{G}_v^{\ *}$)	**8 FPops**		
Step 2.2	*Vdiv*	$\leftarrow$	*compute_div*($\vec{v}$)	**8 FPops**		
Step 2.3	p^{n+1}	$\leftarrow$	*elliptic_solver*(p^n,*Gdiv*,*Vdiv*)	**16 FPops**	$+$	**32FPops/iteration**
Step 3	$\vec{v}^{n+1}$	$\leftarrow$	*correct_velocity*($\vec{v}^n$,p^{n+1},$\vec{G}_v^{\ *}$)	**20 FPops**		
Step 4.1	G_T	$\leftarrow$	*predict_Gt*(T^n,$\vec{v}^{n+1}$)	**39 FPops**		
Step 4.2	T^*	$\leftarrow$	*predict_temperature*(T^n,G_T)	**2 FPops**		
Step 4.3	G_T^*	$\leftarrow$	*correct_Gt*(T^n,T^*,$\vec{v}^{n+1}$)	**79 FPops**		
Step 4.4	T^{n+1}	$\leftarrow$	*correct_temperature*(T^n,G_T^*)	**2 FPops**		
Step 5.1	G_S	$\leftarrow$	*predict_Gs*(S^n,$\vec{v}^{n+1}$)	**35 FPops**		
Step 5.2	S^*	$\leftarrow$	*predict_salinity*(S^n,G_S)	**2 FPops**		
Step 5.3	G_S^*	$\leftarrow$	*correct_Gs*(S^n,S^*,$\vec{v}^{n+1}$)	**79 FPops**		
Step 5.4	S^{n+1}	$\leftarrow$	*correct_salinity*(S^n,G_S^*)	**2 FPops**		
Step 6	ρ^{n+1}	$\leftarrow$	*calculate_density*(T^{n+1},S^{n+1})	**6 FPops**		
Total per time step				**765 FPops**	$+$	**32FPops/iteration**

return($\vec{v}^{final}, p^{final}, T^{final}, S^{final}, \rho^{final}$)

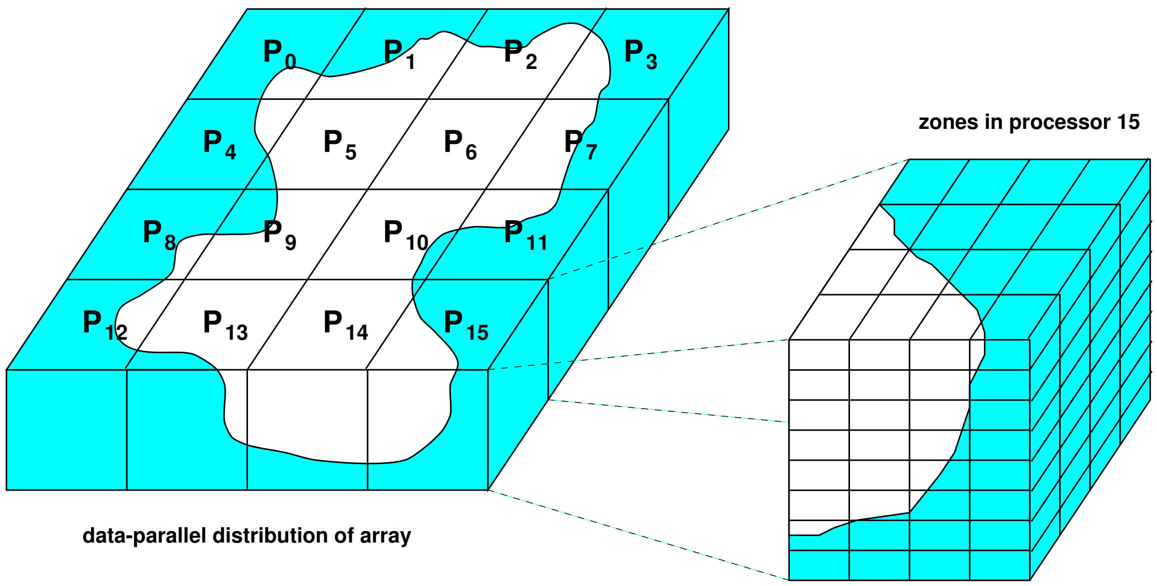


Macro dataflow graph of the main loop


Tree of Activation Frames
main loop
compute Gu
compute Gv
compute Gw
compute u'
compute rho
Gu
Heap Objects



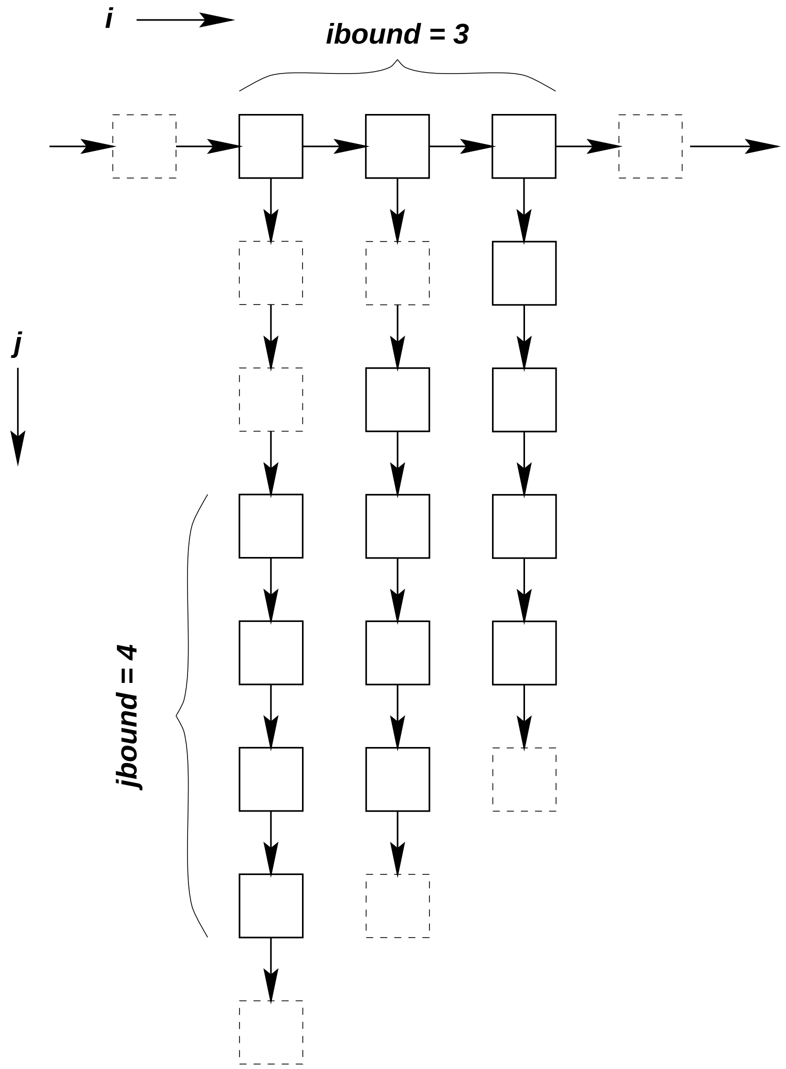
DATA PARALLEL EXECUTION THREAD
Global Synchronization Point
Processor 1
Op: A 1 B 1 C 1
Op: A 2 B 2 C 2
Op: A l B l C l
Processor 2
Op: A l+1 B l+1 C l+1
Op: A l+2 B l+2 C l+2
Op: A m B m C m
Processor 3
Op: A m+1 B m+1 C m+1
Op: A m+2 B m+2 C m+2
Op: A n B n C n
Processor 4
Op: A n+1 B n+1 C n+1
Op: A n+2 B n+2 C n+2
Op: A o B o C o
Identical computations are performed concurrently on every processor. Each processor computes with different data. Modified data is exclusive to the processor that modifies it.
Global Synchronization Point
Processor 1
Processor 2
Processor 3
Processor 4
Data is exchanged ensuring that all processors see the same global memory.
Global Synchronization Point


P0
P1
P2
P3
P4
P5
P6
P7
P8
P9
P10
P11
P12
P13
P14
P15
data-parallel distribution of array
zones in processor 15

Operation	Req. FPOps/ element	Occurrences in elliptic	Cycles per Required Floating Point Op			
			Multithreaded		*Data-parallel*	
			Work	*Work+Idle*	*Raw*	*Adjusted*
inner product	**2**	**4** $\times$	**10.1**		**2.6**	**4.4**
$aX + Y$	**2**	**3** $\times$	**24.4**		**3.2**	**5.4**
7-point stencil	**13**	**1** $\times$	**11.5**		**10.4**	**17.3**
preconditioner	$\sim$ **5**	**1** $\times$	**16.2**		**4.0**	**6.7**
elliptic solver	$\sim$ **32/iter.**		**14.0**	**18.9**	**12.0**	**20.1**
GCM Overall			**13.8**	**14.6**	**9.1**	**15.2**

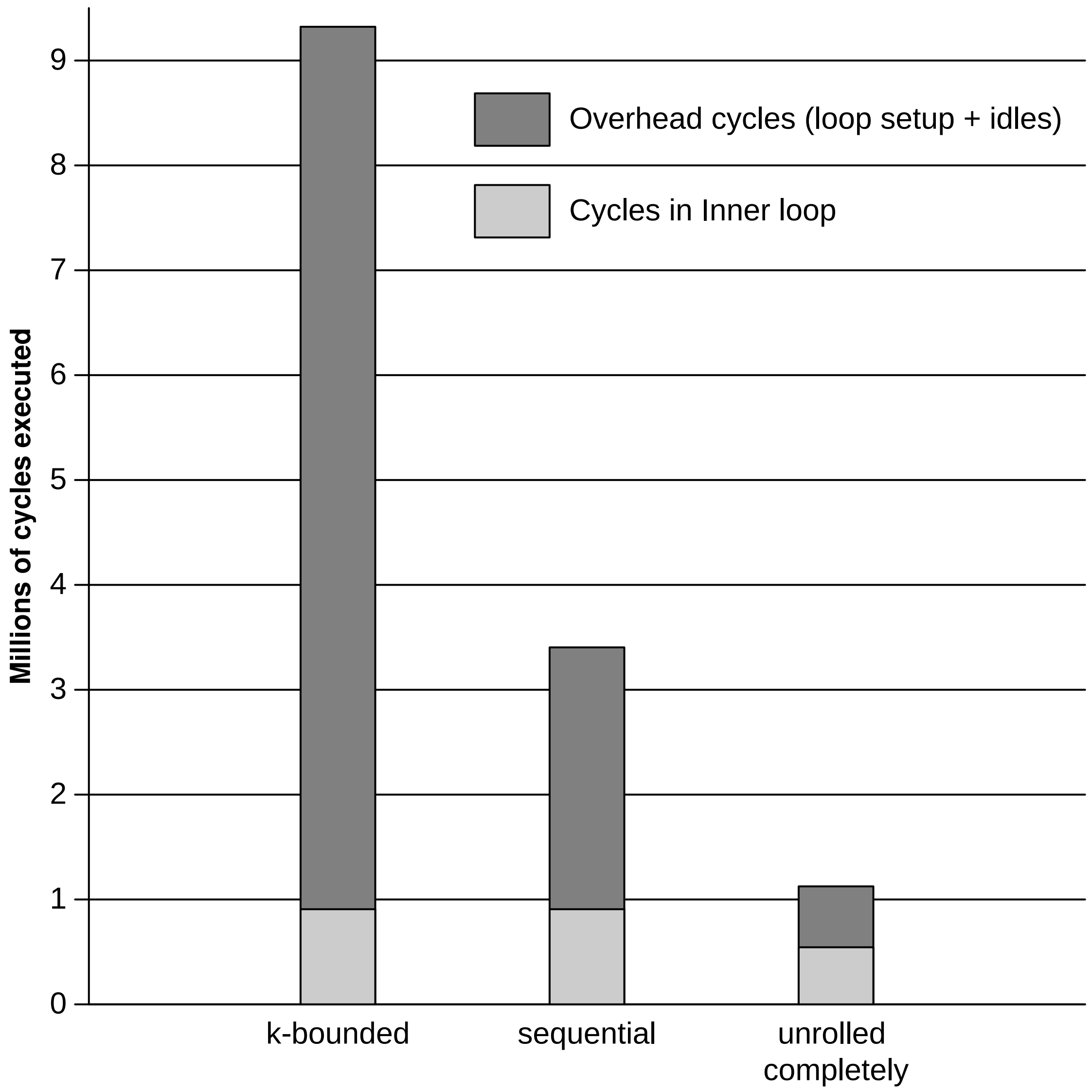
Comparison of performance for the best Id and CM Fortran versions with cycles per required floating point operation, for the $168 \times 90 \times 4$ **Pacific ocean with 60% water. (2 processor Monsoon, 32 processor CM-5). The** *Work* **number for Monsoon includes floating point operations and support instructions. The** *Adjusted* **number on CM-5 is** *Raw* **number** $\div$ **.60.**

```
def ip_3d a b =
  { s = 0.0 ;
   in
     {for i <- 1 to imax BOUND ibound do
        s_jk = 0.0;
        next s = s +
         {for j <- 1 to jmax BOUND jbound do
            s_k = 0.0;
            next s_jk = s_jk +
             {for k <- 1 to kmax do
                next s_k =
                 s_k + a[i,j,k]*b[i,j,k];
                finally s_k};
            finally s_jk};
      finally s} };
```


i
ibound = 3
j
jbound = 4

Type	Loop Schema of Inner Loop (in 1000's cycles) Bounded ($k = 2$)	Sequential	Unrolled
Int Ops	958	444	202
Float Ops	167	137	122
Fetch Ops	618	256	135
Store Ops	126	35	4
Bubbles	1,501	457	185
Move Ops	2,628	1,055	193
Other	3,121	868	102
Work	9,169	3,252	943
Idles	153	151	182
Work + Idles	9,322	3,404	1,125

$ibound = 5$, $jbound = 2$, **2 processor Monsoon**

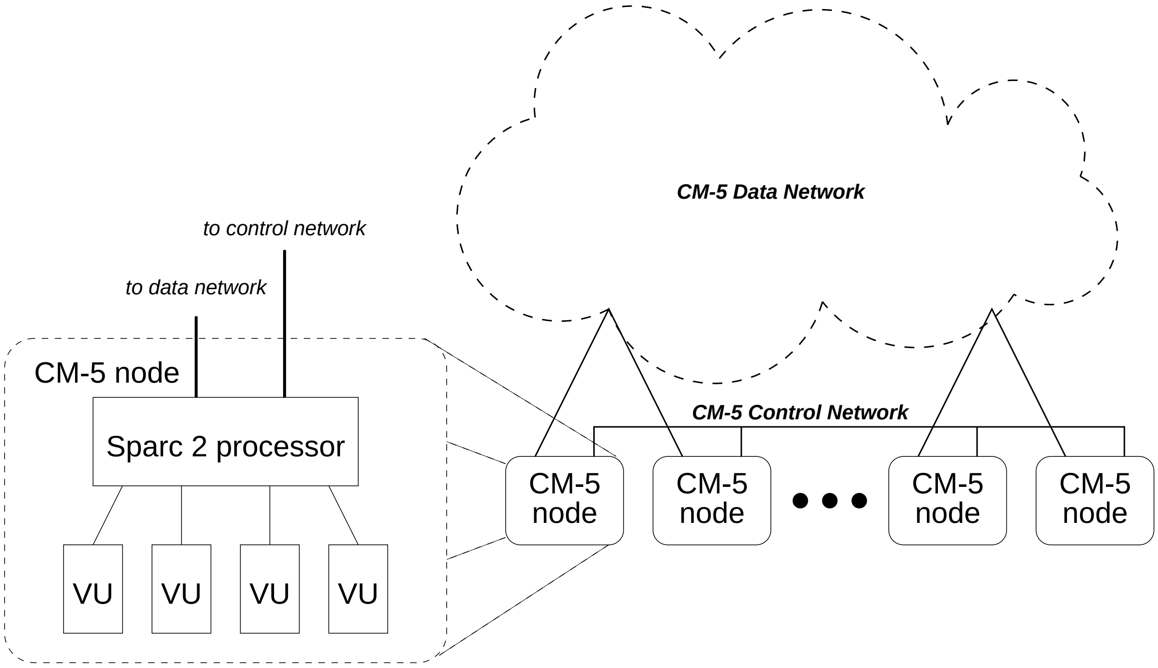

Millions of cycles executed
0
1
2
3
4
5
6
7
8
9
Overhead cycles (loop setup + idles)
Cycles in Inner loop
k-bounded
sequential
unrolled
completely

	One Iteration of Elliptic Solver (in 1000's cycles)		
Type	**Water-Only**	**Everywhere**	**Hybrid**
Int Ops	**6,646**	**3,976**	**3,403**
Float Ops	**1,283**	**1,878**	**1,324**
Fetch Ops	**3,826**	**2,495**	**1,965**
Store Ops	**1,426**	**837**	**764**
Bubbles	**7,692**	**3,357**	**2,893**
Move Ops	**14,677**	**5,293**	**4,397**
Other	**9,718**	**1,445**	**1,514**
Total Work	**45,270**	**19,261**	**16,261**
Idles	**10,975**	**6,321**	**6,321**
Work + Idles	**56,245**	**25,582**	**21,999**
Work / required FLOP	**39.0**	**16.6**	**14.0**
Cycles / required FLOP	**48.4**	**22.0**	**18.9**

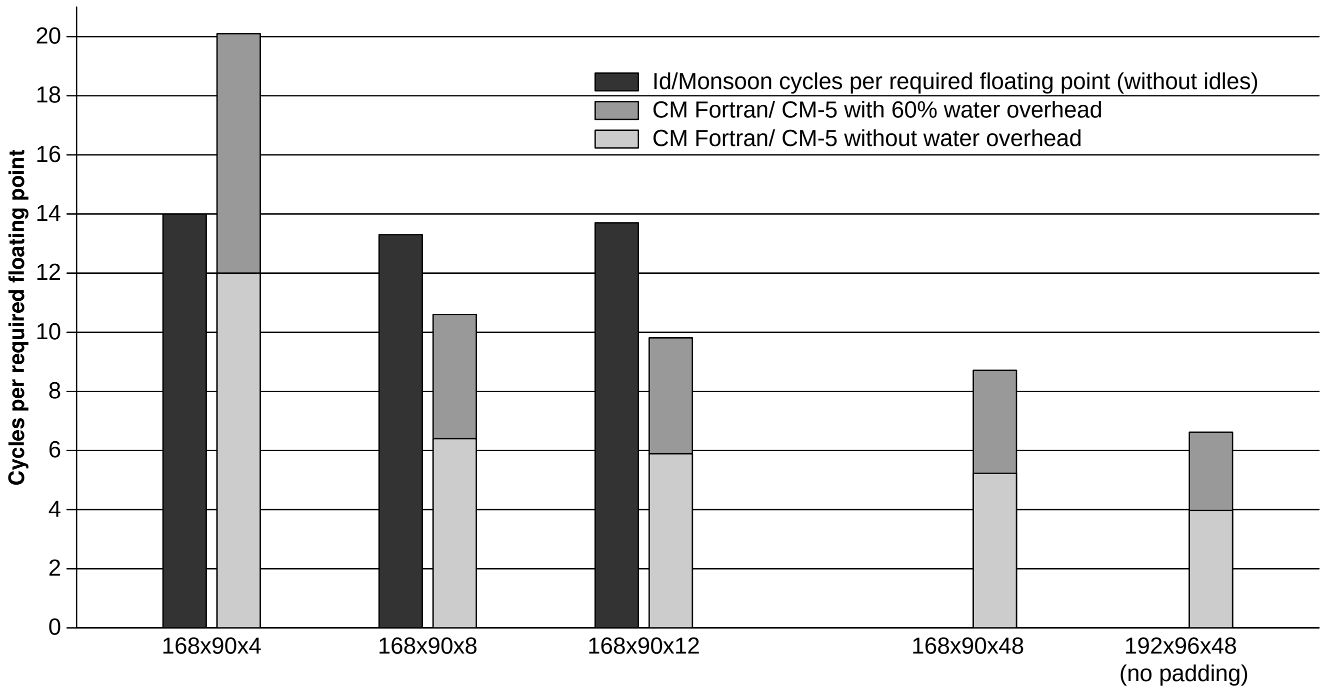
Comparison of the three versions of the elliptic solver for the $168 \times 90 \times 4$ **Pacific ocean. (**$ibound = 5$**,** $jbound = 3$**, 2 processor Monsoon**

Operation	Monsoon Cycles per Required Floating Point Op Computation	Memory	Sync. Overhead Bubble	 Move	Other	*Total* *Work*	Idle	*Total =* *Work+Idle*
inner product	**3.1**	**1.4**	**1.9**	**2.3**	**1.4**	**10.1**	**3.1**	**13.2**
$aX + Y$	**5.5**	**3.1**	**5.6**	**9.0**	**1.1**	**24.4**	**109.3**	**133.7**
7-point stencil	**2.6**	**2.1**	**2.5**	**4.2**	**0.2**	**11.5**	**24.7**	**36.2**
preconditioner	**3.6**	**2.2**	**3.9**	**5.0**	**1.4**	**16.2**	**45.8**	**62.0**
elliptic solver	**4.1**	**2.4**	**2.5**	**3.8**	**1.3**	**14.0**	**4.9**	**18.9**
GCM Overall	**3.8**	**2.3**	**2.6**	**3.6**	**1.4**	**13.8**	**0.8**	**14.6**

Cycles per required floating point operation, for the $168 \times 90 \times 4$ **Pacific ocean, using the** *hybrid* **version of GCM. (**$ibound = 5$**,** $jbound = 3$**, 2 processor Monsoon.)**


CM-5 Data Network
to control network
to data network
CM-5 node
Sparc 2 processor
VU
VU
VU
VU
CM-5 Control Network
CM-5 node
CM-5 node
CM-5 node
CM-5 node

	CM-5 Vector Unit Cycles per Floating Point Operation					
Operation	**Computation**	**Communications**		**Raw Total**	**% Water**	**Adjusted Total**
		Reduce	**CSHIFT**			
inner product	**0.3**	**2.4**	*none*	**2.6**	$\div$ **%60**	**4.4**
$aX + Y$	**3.2**	*none*	*none*	**3.2**	$\div$ **%60**	**5.4**
7-point stencil	**5.4**	*none*	**5.0**	**10.4**	$\div$ **%60**	**17.3**
preconditioner	**4.0**	*none*	*none*	**4.0**	$\div$ **%60**	**6.7**
elliptic solver	**4.1**	**5.2**	**2.7**	**12.0**	$\div$ **%60**	**20.1**
GCM Overall	**7.0**	**0.5**	**1.6**	**9.1**	$\div$ **%60**	**15.2**



Cycles per required floating point
20
18
16
14
12
10
8
6
4
2
0
Id/Monsoon cycles per required floating point (without idles)
CM Fortran/ CM-5 with 60% water overhead
CM Fortran/ CM-5 without water overhead
168x90x4
168x90x8
168x90x12
168x90x48
192x96x48
(no padding)