

pardil (1 call, 87.793 sec)

Generated 08-May-2010 07:25:14 using cpu time.

M-function in file [/Users/jhd/Dropbox/uni/15. semester/msc-vc/svn/msc-vc/code/pardil.m](#)

[Copy to new window for comparing multiple runs](#)

Refresh

- Show parent functions Show busy lines Show child functions
 Show M-Lint results Show file coverage Show function listing

Parents (calling functions)

No parent

Lines where the most time was spent

Line Number	Code	Calls	Total Time	% Time	Time Plot
34	temp = X - (sx * (i-is).*(i-is...	65536	83.203 s	94.8%	
35	output(i,j) = max(temp(:));	65536	4.238 s	4.8%	
36	end	65536	0.341 s	0.4%	
37	end	256	0.010 s	0.0%	
39	end	1	0 s	0%	
All other lines			0 s	0%	
Totals			87.793 s	100%	

Children (called functions)

No children

M-Lint results

No M-Lint messages.

Coverage results

[[Show coverage for parent directory](#)]

Total lines in function	39
Non-code lines (comments, blank lines)	16
Code lines (lines that can run)	23
Code lines that did run	20
Code lines that did not run	3
Coverage (did run/can run)	86.96 %

Function listing

Color highlight code according to

```
time   calls   line  
1 function [output] = pardil(X,scaling)  
2  
1   3   sx = scaling(1);  
1   4   sy = scaling(2);
```

```

5
1 6 if (sx < 0) || (sy < 0)
7     disp('The scaling factor must be >= 0.');
```

8 return;

9 end

10

```
1 11 [m,n] = size(X);
12
1 13 is = zeros(m,n);
1 14 js = zeros(m,n);
15
1 16 for i=1:m
256 17     is(i,:) = i;
256 18 end
19
1 20 for j=1:n
256 21     js(:,j) = j;
256 22 end
23
24 % temp = zeros(m,n);
1 25 output = zeros(m,n);
26
1 27 for i=1:m
256 28     for j=1:n
29 %         for il=1:m
30 %             for jl=1:n
31 %                 temp(il,jl) = X(il,jl) - (sx * (i-il)*(i-il) +
32 %                    sy * (j-jl)*(j-jl));
33 %             end
34 %         end
83.20 65536 34         temp = X - (sx * (i-is).*(i-is) + sy * (j-js).*(j-js));
4.24 65536 35         output(i,j) = max(temp(:));
0.34 65536 36     end
0.01 256 37 end
38
1 39 end
```