

DVC Research Infrastructure: Research Technology Services Optimising Matlab, GPUs and Raijin

John Zaitseff July 2018

# **Optimising your Matlab code**

- 1. Register an account on the Mathworks website
  - https://www.mathworks.com/
  - Free access to documentation, instructional videos, examples, benchmarks and discussion groups
- 2. Read the Matlab documentation
  - Particularly Support » Documentation » Parallel
     Computing Toolbox » Getting Started
- 3. Profile your code
- 4. Check your algorithms
- 5. Vectorise your code
- 6. Use GPUs with gpuArray
- 7. Use parfor, parfeval and spmd
- 8. Talk to others!



als (2) (2) (2) / 1 data 1 john 1 work 1 her and 1	teaching a ser a matish-serallal	
reatt felder	Compand Window	(A) Werkensen
D Nama /	Name to MAY ANY TAXA Data service and for Contrast Hadred	W Nama / Make
devices-alpin-desktop.txt	The statement of the st	Fiant 1x256 double
devices-gamini-desktop.txt	so neight parter in the parallel on univers in parallel and	
devices-raijin-goutst	The centeral form of a parfer statement is:	
devices-raijn-goupascal.txt		
myrand_for.m	<pre>parfor loopvar = initval:endval</pre>	
i myrand gou m	<statements></statements>	
myrand uni en	80	
C uhan factur	MAILAR executes the loop body denoted by STATIONENTS for a vector of	
	iterations specified by INITVAL and EMDERL. If you have Parallel	
	Computing Toolbox, the iterations of STATIMENTS can execute on a parallel	
	pool of workers on your multi-core computer or computer cluster.	
	parter differs from a traditional POR (oop in the following ways)	
	Iterations must be monotonically increasion integer values	
	order in which the loop iterations are executed is not guaranteed	
	Restrictions apply to the STATEMENTS in the loop body	
	parter (Logers = Initvatiendus), RUJ «statementes) RND uses R to	
	spectry the maximum matter of venters in the parallel pool that will	
	By defailt. NULLAR uses as many werkers as it finds available. When	
	there are no workers available in the pool or M is zero, MATLAB will	
	still execute the loop body in an iteration independent order but not	
	in parallel.	
	To order to execute the iterations is excelled, a cacallal seal of	
	workers must arist & narallal and will be created automatically when	
	parter is executed (by default; this can be changed in Settings). A	
	parallel pool can also be created manually using PAMPOOL. PAMPOOL is	
	available with Parallel Computing Teelbox.	
	EVAND E	
	Control Contro	
	Break three large eigenvalue computations across three computers or	
	cores:	
	parter 1 = 113	
	end and	
	See also far, parpool, parallel.Pool, and	
	Reference page for partor	
taits	>> myrand sarfar(1004.1004):	
	Starting parallel pool (parpool) using the 'local' profile	
	connected to 4 workers.	
	Elapsed time is 290.090591 seconds.	
Colorit o file to stars details	so myrand_parter(1004,1024)1	
annex a fee to view decars	Lighted time is 276, SWA22 seconds.	
	Flasted the is 7.35178 seconds.	



### Profile your code

"The real problem is that programmers have spent far too much time worrying about efficiency in the wrong places and at the wrong times; **premature optimisation is the root of all evil** (or at least most of it) in programming."

- Donald Knuth, "Computer Programming as an Art", *Communications of the ACM* 17 (12), December 1974, p. 671, emphasis added.
- Where is your code spending its time?
- Use Matlab's built-in tools
  - profile viewer for a graphical interface
  - tic and toc for basic timing
  - mpiprofile for parallel code
- Focus on areas consuming the majority of time
- Evaluate the effectiveness of your algorithms
- Use built-in functions and toolboxes where possible
  - "Built-in Parallel Computing Support" help topic

<u>File E</u> dit De	bug <u>D</u> esktop	Window	Help						
🔶 🔶 🔶	à 🗚								
Start Profiling	<u>R</u> un this code:	myrand	_for_expand	ed(256,25	6);			<ul> <li>Profile time</li> </ul>	1
myrand_ Generated 26- function in file ( Copy to new wi Refresh I Show parent	for_expand Jul-2018 01:51:15 /data/john/work/hp ndow for companies t functions	ded ( using p oc-eng/ ng mult	Calls: 1 performance teaching/src tiple runs busy lines	., Time time. /matlab-p	e: 7.177 s sarallel/myrand child functions	) for expanded.	<u>m</u>		_
Show Code	Analyzer results 🖬	Show	file coverage	Show 1	unction listing				
Parents (callir No parent	ng functions)								
Lines where	the most time	was sp	ent						1
Line Number	Code	Calls	Total Time	% Time	Time Plot				
<u>9</u>	e = eig(r);	256	7.047 s	98.2%		•			
<u>8</u>	r = rand(A);	256	0.125 s	1.7%	1				
<u>10</u>	ab = abs(e);	256	0.004 s	0.1%					
<u>12</u>	a(i) = m;	256	0.001 s	0.0%					
<u>13</u>	end	256	0.000 s	0.0%					
All other lines			0.001 s	0.0%					
Totals			7.177 s	100%					
Children (calle No children	ed functions)								
Code Analyz	er results								
Line number	Message								
<u>12</u>	The variable ' for speed.	'a'ap∣	pears to c	hange s:	ize on every	loop iterat	ion. Consider prea	llocating	
Coverage res Show coverage	sults a for parent direct	ory							
Total lines in f	unction		13						
Non-code line	s (comments, bla	ank line:	3) 3						



#### Vectorise your code

- Where possible, convert your code from **for** loops to matrix and vector operations
  - Code looks more like mathematical expressions
  - Code is often shorter
  - Code often runs significantly faster
  - Often a prerequisite for good GPU performance
- Example using vectors:

```
i = 0;
for t = 0 : 0.01 : 10
    i = i + 1;
    y(i) = sin(t);
end
t = 0 : 0.01 : 10;
y = sin(t);
```

• Example using arrays:

```
for n = 1 : 10000
    V(n) = 1/12 * pi * (D(n) ^ 2) * H(n)); V = 1/12 * pi * (D .^ 2) .* H;
end
```



## Use parfor, parfeval and friends

- Examine your code for time-consuming for loops and replace with parfor
  - Each iteration must be *independent* (must not depend on the results) of other iterations
  - Reduction variables (e.g., loop summation) are allowed
  - Converting outer for loops work best
  - e.g., Monte Carlo simulations
  - e.g., Parameter sweeps
- Other possibilities: parfeval, distributed, datastore, mapreduce, spmd
- Can use all cores of a multiprocessor compute node
  - On Raijin: **normal** queue up to 16 cores; **normalbw** queue up to 28 cores
- Example:

```
for i = 1 : N
    a(i) = max(abs(eig(rand(A))));
end
parfor i = 1 : N
    a(i) = max(abs(eig(rand(A))));
end
```



# Use GPUs with gpuArray

- Read the "GPU Computing in Matlab" help topic ٠
- Check GPU device capabilities with gpuDevice
- Use **gpuArray** to create arrays on or copy arrays to the GPU
- Use gather to copy arrays back from the GPU
- Check which inbuilt functions can run on the GPU
  - "Run Built-In Functions on a GPU" help topic
  - Currently 344 intrinsic functions
- Profile using tic, toc, gputimeit
- Consider using single-precision calculations
- Example:

```
r_cpu = rand(1024);
r_gpu = rand(1024, 'gpuArray'); % r_gpu array is on the GPU
m = eig(r_gpu);
isOnGPU(m)
```

→ C 🏠 🛈 🚔 https://au.mathworks.co	m/help/distcomp/run-built-i	n-functions-on-a-gpu	uhtmi 🖻 🖪	20% … ☆ 🤇	Search	II\ 📬 🖸 🍕
Documentation			s	Search R2018a Doci	umentation Docu	imentation - C
CONTENTS Close				Trial Software	Product Updates 😗	Translate This Pa
< Documentation Home	abs	compan	flip	isnan	p <g< th=""><th>spdiags</th></g<>	spdiags
( Parallel Computing Toolbox	acos	complex	fliplr	isnumeric	perms	sph2cart
C Parallel computing roolbox	acosd	cond	flipud	isreal	permute	sprand
< GPU Computing	acosh	conj	tioor	ISPOW	pinv	sprandn
< GPU Computing in MATLAB	acot	conv	TPPINT	issorted	planerot	spranosyn
	acotd	conv2	tull	issparse	plot (and related)	spconvert
< Parallel Computing Toolbox	acoth	convn	ganna	1ssymmetric	prus	spnzcart
CODI Computing	acsc	corrcoet	gannainc	1str11	polizcant	sprand
Coro company	acsco	cos	gammalncinv	Istriu	both	sprandn
Pup Built In Eurotions on a CBU	0.000	cosh	gather	koon	polyarea	sprandsyn
run pain-in runctions on a GPU	all	cost	Bacher.	Idiuido	polyder	spriner
ON THIS PAGE	and	cotd	84	lo	polyint	5000070
MATLAB Functions with gpuArray	ana	coth	genes	lenondro	polyint	squeeze
Arguments	angre	coul	graditeric	length	polyral	sublind
Example: Eunctions with anuArray Jacut	arrayfun	cross	hankel	log	now2	subsasen
and Output	aser	rsc	head	10818	nower	subsider
une corpor	asecd	esed	histcounts	login	prod	subspace
Sparse Arrays on a GPU	asech	csch	horzcat	1082	osi	subspect
Considerations for Complex Numbers	asin	ctranspose	hsv2ceb	logical	one	sum
Acknowledoments	asind	cummax	hypot	lsor	ar	superiorfloa
- construction addition and	asinh	cummin	idivide	lt	rad2deg	syd
See Also	assert	cumprod	ifft	lu	rand	svds
	atan	cunsum	ifft2	mat2str	randi	swapbytes
	atan2	deg2rad	ifftn	max	randn	tail
	atan2d	del2	ifftshift	median	randperm	tan
	atand	det	inag	nean	rank	tand
	atanh	detrend	ind2sub	meshgrid	rdivide	tanh
	bandwidth	diag	Inf	min	real	times
	besselj	diff	inpolygon	minus	reallog	toeplitz
	bessely	discretize	int16	mldivide	realpow	trace
	beta	disp	int2str	bon	realsqrt	transpose
	betainc	display	int32	node	rectint	trapz
	betaincinv	dot	int64	novnean	ren	tril
	betaln	double	int8	movstd	repelem	triu
	bicg	eig	interp1	novsum	repnat	true
	bicgstab	eps	interp2	novvar	rescale	typecast
	bitand	eq	interp3	npower	reshape	uint16
	bitcmp	erf	interpn	mrdivide	rgb2hsv	uint32
	bitget	erfc	intersect	ntimes	roots	uint64
	bitor	erfcinv	inv	NaN	rot90	uint8
	bitset	erfcx	ipermute	ndgrid	round	uminus
	bitshift	erfinv	isaUnderlying	ndims	sec	union
	bitxor	exp	isbanded	ne	secd	unique
	blkdiag	expint	iscolumn	nextpow2	sech	uniquetol
	bounds	expm	isdiag	nnz	setdiff	unwrap
	bsxtun	expm1	isempty	nonzeros	setxor	uplus
	cart2pol	eye	isequal	norm	sniftdim	vander
	cart2sph	tactorial	isequaln	normest	sign	var
	cast	Talse	isfinite	not	510	vertcat
	cat	111	1stloat	nthroot	sind	xor
	catzraf	****2	ishermitian	null	single	zeros
	Cell	1111	1510+	nunzstr	sinn	
	cgs	TTESHITE	isinteger	nmet	5120	
	cnoi	Tilter	1510g1cal	ones	sort	_
	circshift	#11ter2	ismatrix	or	SOFTFOMS	~

% r\_cpu array is on the CPU % Do a calculation on the GPU % isOnGPU returns 1 (true)



## **Using GPUs on Raijin**

- Read the GPU User Guide at https://opus.nci.org.au/display/Help/GPU+User+Guide
- Raijin currently has
  - gpu queue: · 30 nodes of four Nvidia K80 accelerators (eight GPUs) each,
     · up to 2.91 teraFLOPS double-precision performance per GPU,
    - up to 8.73 teraFLOPS single-precision performance per GPU,
    - 18 SU (72¢ in-kind contribution) per hour per GPU
  - **gpupascal** queue: · 2 nodes of four Nvidia P100 GPUs each,
    - · up to 5.3 teraFLOPS double-precision performance per GPU,
    - · up to 10.6 teraFLOPS single-precision performance per GPU,
    - $\cdot$  24 SU (96¢ in-kind contribution) per hour per GPU
- Develop code on your workstation or desktop computer
  - Can also use Raijin interactively! Use "ssh -Y" or MobaXterm with inbuilt X server

```
qsub -q gpu -l ngpus=2 -l ncpus=6 -l software=matlab_unsw \
    -l walltime=0:30:00 -l mem=32GB -I -X
```

module load matlab
matlab &



### Why use Raijin? Scale, performance, cost

Desktop computer with Nvidia GTX760: paralleldemo\_gpu\_backslash(0.6)





Raijin K80 node (gpu queue), one GPU: paralleldemo\_gpu\_backslash(6.0)

12000

14000









#### Talk to us!

- John Zaitseff Research Computing Support Engineer J.Zaitseff@unsw.edu.au
- Joachim Mai HPCD Manager Joachim.Mai@unsw.edu.au
- Luc Betbeder-Matibet
   Director, Research Technology Services
   luc@unsw.edu.au



Image credit: UNSW Sydney

Research Technology Services
 DVC Research Infrastructure
 UNSW Sydney
 <u>https://research.unsw.edu.au/research-technology-services</u>

