



*** JACKET RELEASE NOTES AND KNOWN ISSUES ***

10/01/2009 - v1.2 (build 2585)

*** Requires CUDA 2.3 Drivers and Toolkit**

Additions:

- + A Multi-GPU License (MGL) option is now available which adds support for MATLAB's Parallel Computing Toolbox and Distributed Computing Server products for running a single instance of Jacket simultaneously across many GPUs
- + A Jacket MATLAB Compiler (JMC) option is now available which allows you to create standalone executables by compiling Jacket code with the MATLAB compiler, "MCC".
- + bsxfun is now supported for builtin functions (@plus, @times, ...)
- + Functions dependent on bsxfun -
 - cov, std, var are fully supported
 - corrcoef(A) is supported for 2D inputs
- + Preliminary support for sparse GPU matrices:
 - Declaration of sparse GPU matrices as gsingle()
 - Multiplication of sparse matrices with dense matrices
- + Added support for Linear Algebra functions on the GPU using CULA (www.culatools.com)
 - qr and lu supported for real, 2D, gsingle inputs
 - svd support for 2D inputs
 - NOTE: The current implementations of QR and LU are unpivoted
- + mldivide is now supported for real, non-empty gsingle arrays.
- + histograms can now be calculated on gsingle data with hist
- + full is now supported for conversion of sparse matrices to full
- + shiftdim is fully supported

Changes:

- + ginfo(2) now conducts a basic test of your system and prints out result.
- + cache.jkt generated from an older version of Jacket will now not be used in newer releases.
- + gdouple complex support for sum, min, max, any, all, find
- + subscripted references and assignments now support upto 3 GPU variables as indexers, for example:

```
A = gones(4); b = gones(4,1);
A(b)
A(b,b)
A(b,b,b)
```
- + Fixed issue where a lower-class GPU was getting selected by default instead of the one with the most memory, necessitating a gselect operation

- + Fixed bug in permute/ipermute which caused it to fail with [256 256 256] and other similarly-sized inputs
- + Fixed underlying issue that caused GPU failures with subscripted references and assignments beyond a particular index. (A([1:48 100]) is now supported)
- + Updated documentation with GHELP; GHELP without arguments now prints a list of possible arguments
- + gfor i = 1:N now gives a warning that the iterator name may conflict with the MATLAB built-in 'i'
- + sum no longer fails for large array values of the order of 70x50000.
- + fixed an OpenGL initialization issue at startup that caused occasional crashes of the examples

Known issues:

- + Issues with bsxfun and dependents:
 - bsxfun fails with transposed or otherwise reshaped data
 - bsxfun gives incorrect results for the routines xor, geye and mod
 - cov, std, var, corrcoef fail for empty arrays
 - std(A,1,idx) known to fail for certain cases
- + The implementation of svd requires initialization of libraries that conflict with threading libraries initialized by MATLAB at runtime. To force the program to continue, it is necessary to set the environment variable KMP_DUPLICATE_LIB_OK to TRUE. This can be done on a (Linux) bash shell as:


```
export KMP_DUPLICATE_LIB_OK=TRUE
```

 Currently, the Windows installation is configured to automatically set this value at runtime.
- + Please read the Technical Notes on svd and mldivide by typing 'GHELP SVD' and 'GHELP MLDIVIDE' on the MATLAB console for information on comparing the CPU and GPU-generated values of these functions. You may also visit the FAQ page at www.accelereyes.com/faq.php for more information.
- + Currently grand without arguments returns an error instead of defaulting to grand(1)

08/12/2009 - v1.1.1 (build 2057)

*** Requires CUDA 2.3 Drivers and Toolkit**

Additions:

- + fft is now supported for gdouble inputs.
- + grandn - GPU version of randn.
- + grand - GPU version of rand.
- + sort(X) is now supported.
- + cumsum(X) is now supported for vectors.
- + cumprod(X) is now supported for vectors.
- + permute, ipermute is now supported.
- + repmat is now supported for complex inputs.
- + The gcache command now allows users to manipulate the cache by flushing, saving or loading it.
- + The command 'ghelp' displays information about functions supported by Jacket
 - on the MATLAB console:
 - Features different from MATLAB's implementation of the function
 - Supported syntax for the function
 - Where relevant, whether a function can be used within a gfor loop.
- + mod and rem now have complex support.
- + xor support for float data types.

Changes:

- + conv2(A,B,'same') with subscripted references to A and B now produces correct results.
- + dot(A,B) performed on 2D matrices now gives an error message instead of an incorrect result.
- + Fixed a rounding issue that caused cast to work incorrectly on gsingle-to-guint32 conversion.
- + isinteger now returns a single value for an array, instead of an array of values.
- + Fixed an issue in isfloat which was causing it to return incorrect results.
- + Sign no longer gives incorrect results for real gdouble data.
- + Fixed an issue where mod did not work similar to MATLAB.
- + Fixed an issue where rem returned incorrect results.
- + Subscripted assignments of the type A(B) = C now work for real A and complex C.
- + circshift no longer gives erroneous results with shifts greater than the dimension of the input.
- + atan2 and hypot no longer cause GPU failure for real inputs of different precision types.
- + repmat(A,M,N) now returns an error message if M and N are GPU variables.
- + Fixed a bug that caused assignments of the type A = B(1:N,:) to crash beyond a particular subscript N.

Known Issues:

- + FFT(A) where A = grand(1), may produce incorrect results.
- + Filter(B,A,X) is known to crash with inputs of large dimensions.
- + complex returns incorrect values when run on scalars.
- + cat(dim,A,B) fails when concatenating large-dimensional inputs.
- + sum produces incorrect results with large matrices (on the order of 100x50000).
- + b=a([1:48 100]) causes a GPU failure.
- + find does not support complex data - currently it returns incorrectly sized results.

06/17/2009 - v1.1 (build 1456)

*** Requires CUDA 2.2 Drivers and Toolkit**

Additions:

- + gdouble support.
- + other new native GPU data types supported: glogical, gint32, guint32.
- + new developer SDK available offering both designated computer and concurrent network licensing options.
- + partial support for conv2: the method conv2(a,b,'same') is now supported for kernels of any size up to 5x5 (e.g. 1x3 2x5 3x3 4x5 ...) and square kernels from n=6 to n=10 (e.g. 6x6 7x7 8x8 9x9 10x10).
- + new licensing scheme for designated computer and concurrent network licenses.
- + ginfo now detects the presence (or absence) of CUDA and gives an appropriate message.

- + colon is fully supported (except in gfor).
- + trace() supported for complex matrices.
- + repmat(A,M,N) is fully supported. repmat(A,[M,N,P . . .]) is supported for up-to 3-dimensional inputs and 3-dimensional outputs.
- + the reductions - sum, min, max, any, all are now supported.
 - preliminary gfor support for reductions is now present.
 - reductions can also be performed on complex numbers.
 - note: reductions for complex gdouble is not supported.
 - min and max now give indexed output over real-valued vectors. This support is currently enabled only for gsingle arrays.

Changes:

- + the method max(A,B) now produces consistent results when performed on complex gsingle variables.
- + subassigning a gdouble to a gsingle no longer causes an error.
- + fixed scaling issues at large data sizes in IFFTn.
- + fixed failure when /tmp is not writable.
- + files written to /tmp no longer have reproducible filenames.
- + "clear all" clears all GPU memory. After this, ginfo will report 0mb used.
- + diag now supports gfor.
- + Fixed issue related to subscripting that caused failure on Jacket:

```
>> A=gonex(5,5,20);
>> B=A;
>> A(:,:,2)=B(:,:,2);
>> A(:,:,2)
```
- + multiplication of an array (of size > 1) of gingles that have been labelled as 'logicals' now produces consistent results, e.g (gonex(2)>0)*gsingle(5).
- + subassigning a gdouble to a gsingle is now supported.
- + max(A,[],dim) is supported for up to 3D arrays over 3rd dimension.
- + meshgrid now supports mixed types and mixed complexity; this means meshgrid(gsingle,gdouble) is allowed. However it does not support mixing logical and floating point.
- + find now supports indices in 2 coordinate form.
- + various other fixes to problems reported on the forums.

Known issues:

- + FFT is only single-precision and will become double-precision once CUFFT gains double-precision support.
- + gfor will abort if used within nested function.
- + fft as well as ifft supports results only along the first non-singleton dimension. fft(A,N,dim) gives inconsistent results.
- + result of fft2 followed by ifft2 on GPU gives a small img component. This is a precision issue and will be resolved when double-precision FFT are supported.
- + array rearrangement using circshift causes crashes/out of memory in some cases with large array sizes.
- + crash on the following subsref case: a=gonex(1,100); b=a([1:48 100]).
- + subscripted assignment into an unallocated matrix is not supported yet:

```
clear A; A(:,:,1) = gonex(3,3);
```

05/15/2009 - v1.0.3 (build 839)

*** Requires CUDA 2.2 Drivers and Toolkit**

*** Primarily a stability release in preparation for gdouble**

Additions:

- + colon operator
- + subsasgn: complex data

Changes:

- + major upgrades to general and FFT memory subsystems
- + major big fixes throughout system (gfor, subsasgn, ...)

Known issues:

- + gfor will abort if used within nested function

04/11/2009 - v1.0.2 (build 620)

*** Requires CUDA 2.1 Drivers and Toolkit**

Additions:

- + [A A] and [A; A] (horzcat, vertcat, cat)
- + gamma, gammaln
- + find - support for returning: index, row/column, values
- + gcache - export your compiled GPU code for use in later sessions

Changes:

- + various fixes from the forums
- + gfor - various fixes for subscripting and iterator manipulation
- + handling empty matrices

02/02/2009 - v1.0.1 (build 541)

*** Requires CUDA 2.1 Drivers and Toolkit**

Additions:

- + gfor: supports fft, fft2, fftn and their inverses

Changes:

- + bug fix: fft/iffn correctly operates on ND data
- + bug fix: gactivate is more robust to manually edited jlicense.dat files

01/26/2009 - v1.0 (build 513)

*** Requires CUDA 2.1 Drivers and Toolkit**

Additions:

- + gfor/gend (preliminary): simultaneous for-loops on the GPU
- + trace saving: avoid runtime kernel compilations with this new feature
- + convn: only for 3x3x3 kernels and 3D volumes
- + fftshift/iffnshift support
- + interp2 support
- + subsreferencing into logical data (still cannot subsasgn)
- + a new FDTD example (used in electromagnetics and seismic applications)
- + a new Monte-Carlo integration simulation example with gfor
- + new gfor examples demonstrating matrix multiplication and FFT computations

Changes:

- + ginfo reports memory as: CPU-used, GPU-used, GPU-free
- + bug fix: fftn/iffn memory
- + bug fix: graphics on Mac OSX

- + bug fix: transpose now works on all datatypes (logicals,float,uint32,int32)
- + bug fix: fft2 batch processes higher dimensions

11/24/2008 - v0.6 (build 328)

*** Requires CUDA 2.0 Drivers and Toolkit**

Additions:

- + Jacket now runs over SSH (for Linux SSH servers only)
- + sum/min/max/any/all: along any dimension of N-dimensional data
- + filter now supports input/output conditions
- + fftn,ifftn supporting 1D,2D,3D (optional padding)
- + subscripting on logical data
- + gscatter3 - Graphics Toolbox version of scatter3
- + gpatch - Graphics Toolbox version of patch

Changes:

- + better handling of memory pressure and associated error messages
- + bug fix: issues relating to sub-referencing using gsingles
- + bug fix: issues relating to sum on small vectors
- + bug fix: issues relating to the handling of the glColorPointer and glNormalPointer calls in the Graphics Toolbox

09/19/2008 - v0.5 (build 215)

*** Requires CUDA 2.0 Drivers and Toolkit**

Additions:

- + Graphics Toolbox for all operating systems
- + Custom MEX support - now you can write your own CUDA kernels to run on gsingles (see the mex_example)
- + filter (IIR)
- + circshift
- + new graphics examples: gfx_raindrop_example, gfx_shaded_ocean_example, gfx_interactive_ocean_example

Changes:

- + faster filter2 (now uses native 3x3 MATLAB kernels)
- + faster subscript references
- + sum/min/max/any/all: along first dimension of N-dimensional data
- + bug fix: reshaping after subscripting
- + bug fix: examples now use single-precision on the CPU-side

Known bugs:

- + sum() sometimes fails when summing lots of small vectors

08/14/2008 - v0.4

*** Requires CUDA 2.0 Beta 2 Drivers and Toolkit**

Changes:

- + fixed find() - no longer uses CUDPP
- + gforce properly blocks until all GPU computation finishes
- + subscripting along columns of matrix or slices of a volume is very efficient (or any contiguous span of a matrix)
- + gsingle allowed in subscript references
A = gsingle(1:5);

```

    B = gsingle([1 2 4]);
    A(B)
+ gsingle allowed in subscript assignment
    A = gsingle(rand(1,5));
    B = gsingle([1 2 4]);
    A(B) = exp(B);
+ logicals allowed in subscript assignment with a scalar assignment
    A = gsingle(rand(1,5));
    A(A > .5) = 0;
+ ginfo reports data memory in use and available (not total card
memory), reports current operating frequency
+ help documentation: gones, gzeros, ginfo, gforce, geye,
gsingle
+ logical operations use less memory (1 byte booleans rather than 4
byte floats)
+ gfor has been removed from this release but expect it back soon

```

Known Issues:

```

+ logical expressions are still being developed, for example:
+ cannot subscript assign a gsingle from a logical expression:
    A = gones(3);
    A(1:3) = [true true true];
+ cannot subscript reference a gsingle from a logical expression
    A = gones(3, 'logical');
    A(1:3)
+ using a gsingle in cell array is unstable

```

07/10/2008 - v0.3 - Many performance enhancements. Changed 'gwhos' to 'ginfo'. Added advanced functions, such as 'gfor' and 'gforce'. New functionality, such as filter2 for 3x3 kernels. Alpha graphics extension included in the linux build.

06/28/2008 - v0.2 - Updated the MAC build, fixing several small issues.

06/25/2008 - v0.2 - Fixed several issues with the Windows build. Did not update release date.

06/21/2008 - v0.2 - Fixed several small bugs and add a few performance enhancements.

06/18/2008 - v0.2 - First beta releases