MathWorks®

MATLAB Answers

When to use codistributed arrays

⊕Follow

8 views (last 30 days)

Daniel Jaló on 12 Feb 2013

Accepted Answer: Jill Reese **STAFF**

• 👍Vote

0

Imagine I have the following matrix:

A = rand(6400,6400)

Now imagine I create a distributed array from it: (I have 4 workers)

dist = codistributor1d();
dist = codistributor2dbc([2 2],3200);
B = codistributed(A,dist)

There are two ways for me to distribute it. Either each worker stores a 3200x3200 matrix or each worker stores a 6400x1600 matrix.

My questions are:

• When should I distribute an array?

• How do I know which function, codistributor1d or codistributor2dbc, I should use whenever I have some array I want to distribute between workers? I know how to work with both type of arrays but I don't know when one is better than the other.

If anyone could help me I'd appreciate.

💬 **0 Comments**

Sign in to comment.

Sign in to answer this question.

✅Accepted Answer

Jill Reese **STAFF** on 14 Feb 2013

• 👍Vote

0

• 🔗Link

Distributed arrays are most useful when you do not have enough memory to store an entire array on a single machine. By distributing chunks of the original array across all the workers in the pool, you can perform operations on the entire array that you previously couldn't even store. I would suggest that you take a step back and start by working with distributed arrays rather than codistributed arrays. This allows MATLAB to choose a default distribution scheme for you.

MATLAB Answers

A = rand(6400,6400);

matlabpool open

dA = distributed(A); % let MATLAB pick a default distribution scheme for the

% distributed array, dA

R = chol(dA); % example of a function that works for

% distributed arrays - no spmd required

With distributed arrays, you can get started without having to worry about what distribution scheme to use. If you are interested, you can always query the distribution scheme that is currently used by a distributed array like so:

% dA is the distributed array created as above

spmd

codistr = getCodistributor(dA) % inside spmd we can "view" dA as a

% codistributed array and access its

% distribution scheme

end

Once your code is working correctly with distributed arrays, you can make minor changes to use codistributed arrays with a specific distribution scheme. Changing the default distribution scheme may improve performance, but as Matt J mentions, choosing the most efficient distribution scheme is problem dependent and will depend heavily on the operations that you want to call.

💬 **0 Comments**

Sign in to comment.

💬 More Answers (1)

MJ

Matt J **MVP** on 12 Feb 2013

Edited: Matt J **MVP** on 12 Feb 2013

- 👍 Vote

0

- 🔗 Link

How you cut up the array would depend on what portions of your array each parallel job needs. If each parallel job requires all rows of A, but not all columns, it makes more sense to split as 6400x1600 than as 1600x6400. That way all data needed for the job will be available on the associated lab.

💬 **0 Comments**

Sign in to comment.

Sign in to answer this question.

See Also

MATLAB Answers

Distributing arrays to workers for local processing ?

1 Answer

-maxNumCompThreads, hyperthreading, and parpool

2 Answers

Non distributed arrays in SPMD

1 Answer

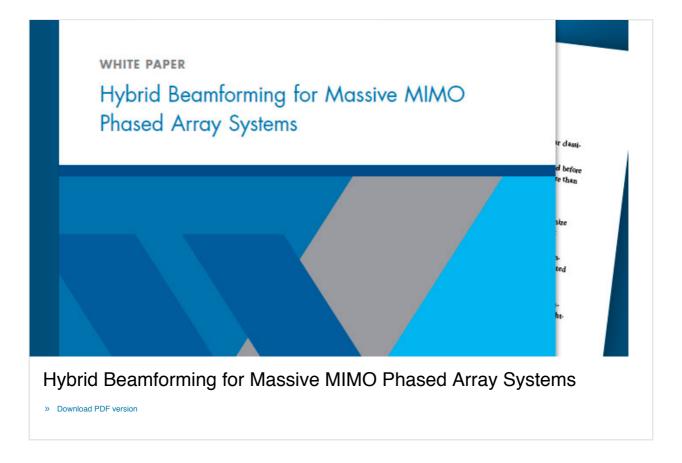Entire Website

Profiling Load Unbalanced Codistributed Arrays

Documentation

codistributed.sprandn

Documentation

Working with Codistributed Arrays

Documentation

MATLAB Answers

Categories

Parallel Computing > Parallel Computing Toolbox > Big Data Processing > Distributed Arrays

Tags

parallel   parallel computing   codistributed arrays   distributed   codistributor

Products

Parallel Computing Toolbox

WHITE PAPER

Hybrid Beamforming for Massive MIMO Phased Array Systems

Hybrid Beamforming for Massive MIMO Phased Array Systems

» Download PDF version

mathworks.com