

Da: Peter Hunter <p.hunter@auckland.ac.nz>
Oggetto: **Re: PDE 2 ODE via CUDA**
Data: 06 luglio 2010 18.02.36 GMT+02.00
A: Andrew McCulloch <amcculloch@ucsd.edu>
Cc: "emanuela.merelli@unicam.it" <emanuela.merelli@unicam.it>,
"luca.tesei@unicam.it" <luca.tesei@unicam.it>, Andrew Miller
<ak.miller@auckland.ac.nz>, Marco Viceconti <viceconti@TECNO.IOR.IT>,
Johan Hake <jhake@bioeng.ucsd.edu>, Fred Lionetti <flionetti@gmail.com>,
Jeff Van Dorn <jvandorn@ucsd.edu>, Poul Nielsen
<p.nielsen@auckland.ac.nz>

Great, thanks Andrew! Good travels.
Peter

On 7/07/2010 3:57 a.m., Andrew McCulloch wrote:

I'm about to leave town but I will ask Fred Lionetti to send a copy of his translator code and paper to Andrew. Note that Fred now works for a company in Washington State so he may not be able to get to this immediately. Jeff Van Dorn may be able to help out in the meantime.

regards,

Andrew

On Jul 6, 2010, at 2:36 AM, Peter Hunter wrote:

Thanks Andrew. Look forward to ms and we'll keep you posted if/when we get underway with a CUDA API. Next step is probably for Andrew (Miller) to try out your CUDA ODE solver - presuming it is free open/source ??.

Cheers, Peter

Sent using BlackBerry

----- Original Message -----

From: Andrew McCulloch<amcculloch@ucsd.edu>

To: Peter Hunter

Cc: emanuela.merelli@unicam.it<emanuela.merelli@unicam.it>;

luca.tesei@unicam.it<luca.tesei@unicam.it>; Andrew Miller;

viceconti@TECNO.IOR.IT<viceconti@TECNO.IOR.IT>; Johan Hake<jhake@bioeng.ucsd.edu>

Sent: Tue Jul 06 16:34:58 2010

Subject: Re: PDE 2 ODE via CUDA

Hi Peter,

We do have a C source to Cuda source translator and ODE solver recently accepted for publication at EuroPar 10. I can send the ms later this week. It includes various GPU specific optimizations some of which are more effective than others. In Continuity we generate the C

automatically from SymPy. Johan Hake in my group has expressed interest in writing a CellML to SymPy translator and has some useful preliminary progress.

Regards,

Andrew

Sent from my iPhone

On Jul 5, 2010, at 12:14 AM, Peter Hunter<p.hunter@auckland.ac.nz> wrote:

Hi Andrew,

Would you mind having a look through the correspondence below. Who is the best person in your group to talk to about a CellML CUDA link?

Cheers,

Peter

Sent using BlackBerry

----- Original Message -----

From: Marco Viceconti<viceconti@TECNO.IOR.IT>

To: Andrew Miller

Cc: Peter Hunter; Merelli Emanuela<emanuela.merelli@unicam.it>; Luca

Tesei<luca.tesei@unicam.it>

Sent: Mon Jul 05 18:18:50 2010

Subject: Re: PDE 2 ODE via CUDA

What I found so far is that there is a cuda port of LSODA:

<http://sourceforge.net/projects/culsoda/>

I also noticed a presentation from Andrew McCulloch group where they report some preliminary results for a CUDA ODE solver. Should we contact him to see where they are?

Marco

Il giorno 4 Jul 2010, alle ore 22:45, Andrew Miller ha scritto:

On 05/07/10 04:11, Peter Hunter wrote:

Hi Andrew,

Would you mind reading below and giving me your thoughts re a CellIML API for CUDA - espec how much work you think there is involved in doing this.

Hi Peter,

It is hard to say from the information provided. The CellIML API has a service called CeLEDSExporter (CellIML Language Export Definition Service) which takes an XML description of a language, and generates procedural code. It is quite possible that it would be possible to make an XML-file describing how to transform CellIML to C for CUDA without needing to make any actual code changes to the CellIML API.

To do this, however, we are going to need to find a good numerical solver library that works on CUDA - we currently use CVODE, GSL, and IDA, and as far as I can tell, none of those have been adapted to work on CUDA. If we can't find one that works out of the box, then it will be a case of porting or writing a good numerical solver on CUDA, which will be a lot of work.

Best wishes,
Andrew

Marco& Emanuela: This is the first Andrew has heard about this so we need to be cautious about expectations until Andrew has had a chance to assess the workload.

Cheers,
Peter

----- Original Message -----

Subject: Re: PDE 2 ODE via CUDA
Date: Mon, 5 Jul 2010 04:01:37 +1200
From: Marco Viceconti<viceconti@tecno.ior.it>
To: Peter Hunter<p.hunter@auckland.ac.nz>
CC: Catherine Lloyd<c.lloyd@auckland.ac.nz>, Pani Martino <pani@tecno.ior.it>, Taddei Fulvia<taddei@tecno.ior.it>, Luca Tesei <luca.tesei@unicam.it>, Merelli Emanuela<emanuela.merelli@unicam.it>

I leave this to Emanuela, to me it sounds good, but we need to have something running quite soon, so I would suggest Emanuela and Luca talk to Andrew and see how this might work out.

I am already in Edinburgh, I lectured at the pre-course this morning, and now I am stuck in my last meeting + dinner of the ESB council (stepping out after eight years). I shall see you at your lecture tomorrow morning.

Cheers

Marco

Il giorno 4 Jul 2010, alle ore 15:39, Peter Hunter ha scritto:

Hi Marco,

We too have thought that this would be a good approach (e.g. for OpenCMISS) and plan to implement a CellML API to generate CUDA code (currently on the website you can display generated code in C, F77, Matlab and Python). I'll check with the guys in Auckland as to how long it would take to do this. Maybe a good strategy would be to put Emanuela in contact with Andrew Miller in Auckland who would be the one implementing the CellML API for CUDA. If this sounds like a good approach, I'll email Andrew.

BTW, I'm in Edinburgh now for ESB2010 - when do you arrive here?

Cheers,
Peter

On 4/07/2010 11:55 p.m., Marco Viceconti wrote:

Dear Peter:

we recently started an exploratory collaboration with a computer science group in Italy lead by Emanuela Merelli, on the possibility to couple our PDE of boen tissue biomechanics, with ODE remodelling models such as those Catherine has been kindly annotating in CellML recently. Since at each time step there are around 10^6 cells that remodel, the problem is to solve so many copies of the ODE model simultaneously, and this can be done with CUDA.

Recently Emanuela showed us that they were able to solve more than one million ODE with a single time-step Euler in seconds. Assuming a some dozen of time steps, this would mean we can solve one million ODE in minutes.

As next step we need to implement on the CUDA architecture an ODE solver generic enough to cope with various problems, so as to ensure

the work we are doing can be reused elsewhere. Our idea is at one point to be able to pick the CellML model, dump it onto the CUDA machine, and run many instances of it with different initial values.

We were looking at libraries, but before we move on, I thought that if

we could use the same code you use in OpenCell, we could ensure that

everything run under OpenCell should run identical under the CUDA system, which seems a nice feature, given that OpenCell is probably the tool we shall use to develop and test CellML models.

I would love to have your meaning on this idea, and some extra information (in case you agree with it) on how we could practically do, and from which software artifact we could start from.

Cheers

Marco

MARCO VICECONTI, PhD

(viceconti@tecno.ior.it)

Laboratorio di Tecnologia Medica

tel. 39-051-6366865

Istituto Ortopedico Rizzoli

fax.

39-051-6366863

via di Barbiano 1/10, 40136 - Bologna, Italy

Tiger! Tiger! Burning bright in the forest of the night,
what immortal hand or eye could frame thy fearful symmetry?

Opinions expressed here do not necessarily reflect those of my employer

MARCO VICECONTI, PhD

(viceconti@tecno.ior.it)

Laboratorio di Tecnologia Medica

tel. 39-051-6366865

Istituto Ortopedico Rizzoli

fax.

39-051-6366863
via di Barbiano 1/10, 40136 - Bologna, Italy

Tiger! Tiger! Burning bright in the forest of the night,
what immortal hand or eye could frame thy fearful symmetry?

Opinions expressed here do not necessarily reflect those of my
employer

MARCO VICECONTI, PhD
(viceconti@tecno.ior.it)
Laboratorio di Tecnologia Medica tel. 39-051-6366865
Istituto Ortopedico Rizzoli fax.
39-051-6366863
via di Barbiano 1/10, 40136 - Bologna, Italy

Tiger! Tiger! Burning bright in the forest of the night,
what immortal hand or eye could frame thy fearful symmetry?

Opinions expressed here do not necessarily reflect those of my employer

Andrew McCulloch
amcculloch@ucsd.edu