International *Mathematica* User Conference 2009

Delegating Computations to C Code

or How to make *Mathematica* even Faster ?

Yves A. Papegay

INRIA Sophia Antipolis - COPRIN

Motivations

Several basic assumptions

Mathematica is universal

- data
- paradigms

Mathematica is powerful and fast

Mathematica is simple and easy to use

Mathematica is a small piece of C code ... and a huge piece of *Mathematica* code

One immediate consequence

I love using *Mathematica*

But I have to pay for that ...

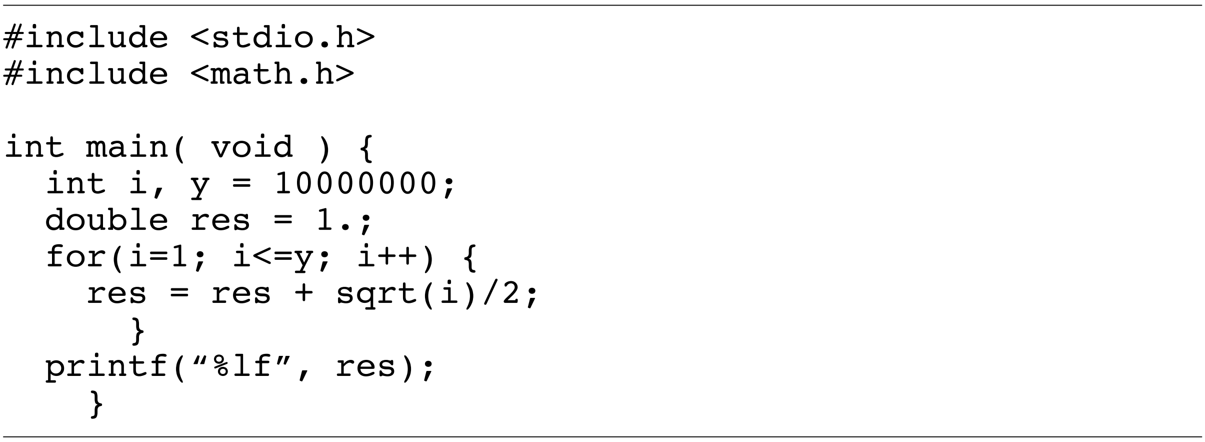
- ... in term of performances

Motivations - an exemple

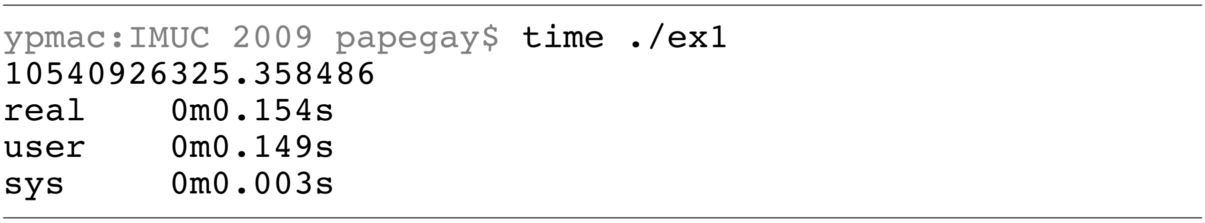
Mathematica

```
f[x_, y_] := N[Fold[(#1 + Sqrt[#2] / 2) &, x, Range[y]]]

Timing[f[1, 5000]]

Timing[f[1., 10 000 000]]

g = Compile[{{x, _Real}, {y, _Integer}}, f[x, y]]

Timing[g[1, 10 000 000]]

Timing[g[1., 10 000 000]]
```

C

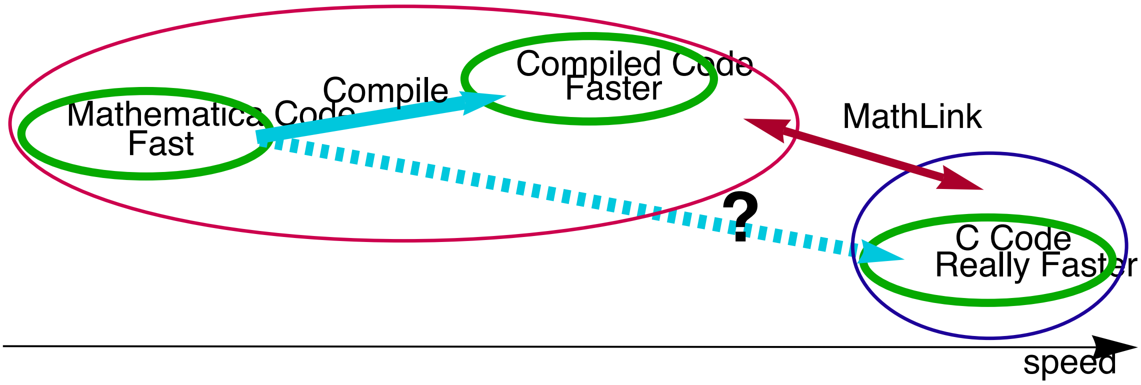


```
#include <stdio.h>
#include <math.h>

int main( void ) {
  int i, y = 10000000;
  double res = 1.;
  for(i=1; i<=y; i++) {
    res = res + sqrt(i)/2;
      }
  printf(“%lf”, res);
    }
```

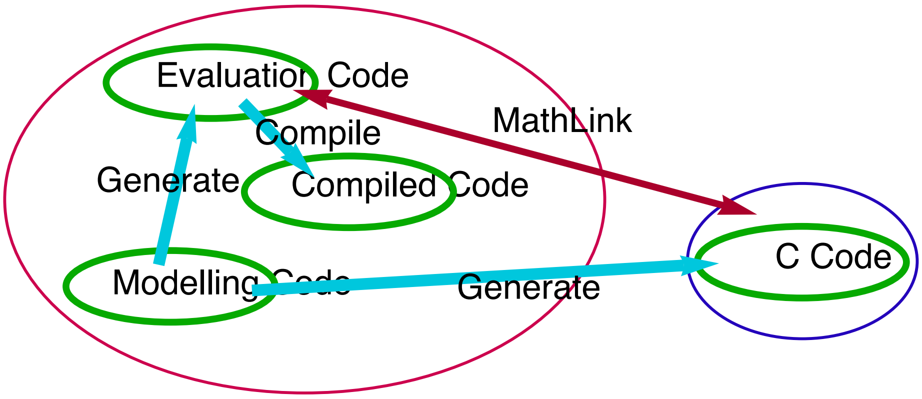


```
ypmac:IMUC 2009 papegay$ time ./ex1
10540926325.358486
real     0m0.154s
user     0m0.149s
sys      0m0.003s
```

Delegating to C Code



Modelling, delegating to C Code



Modelling Code

- declarative langage embedded in a functional form (with inert functions)

```
ModelEx[{
  Inputs[v1, Integer],
  Inputs[v2, Real],
  Inputs[v3, Real],
  Formula[v4, v2^v1],
  Formula[v5, v4*v3[-1]],
  Formula[v6, If[v4[-1] > v4, v5 - v2, v5]],
  Output[v6] }]
```

```
"show a less simple example !"
```

```
model
```

Evaluation Code

```
ModelEx[t_] :=
 ModelEx[t] = Module[{
     v1 = INPUT[1, t], v2 = INPUT[2, t], v3 = INPUT[3, t],
     v3p = INPUT[3, t - 1],
     v4, v4p = ModelEx[t - 1][[2, 1]],
     v5, v6},
    (v4 = v2 ^ v1;
     v5 = v4 * v3p;
     v6 = If[v4p > v4, v5 - v2, v5];
     {v6, {v4}})]
```

```
ModelEx[{
  Inputs[v1, Integer],
  Inputs[v2, Real],
  Inputs[v3, Real],
  Formula[v4, v2 ^ v1],
  Formula[v5, v4 * v3[-1]],
  Formula[v6, If[v4[-1] > v4, v5 - v2, v5]],
  Output[v6] }]
```

show a less simple example !

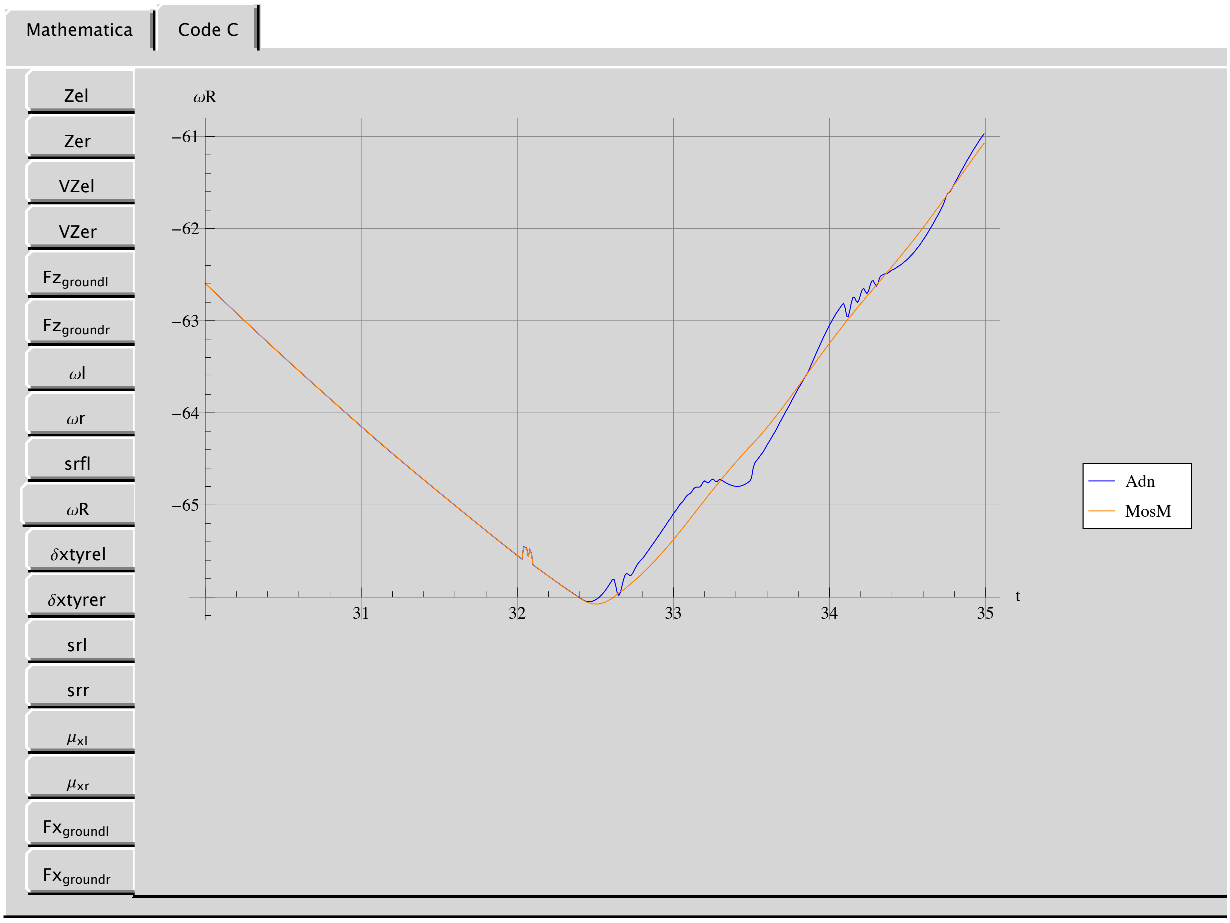
C Code

show a less simple exemple !

```
CCode[ProgramC[
  IncludeC["stdio.h"], IncludeC["math.h"], EolC,
  {DeclareC["double", ArrayC["modelout", "MAXT"]], DeclareC["double", "modelmemory"], EolC},
  FunctionMC["void", "model", {{"int", t}},
   Join[{DeclareC["int", v1], DeclareMC["double", {v2, v3, v3p, v4, v4p, v5, v6}], EolC},
    MapIndexed[AssignC[#1, CallC["readInput", {#2[[1]], t}]] &, {v1, v2, v3}],
    {AssignC[v3p, CallC["readInput", {3, t - 1}]],
     AssignC[v4p, "modelmemory"]},
    {AssignC[v4, v2^v1],
     AssignC[v5, v4 v3p],
     IfThenMC[v4p > v4, {AssignC[v6, v5 - v2]}, {AssignC[v6, v5]}]},
    {AssignC[ArrayC["modelout", t], v6], AssignC["modelmemory", v4]}]]]]

codec
```

Performances

C code is between 25 and 100 times faster (here 500 steps in 28 s. vs. 0.32), Compile give a a gain of 30% to 50%


Mathematica
Code C
Zel
Zer
VZel
VZer
Fz$_{groundl}$
Fz$_{groundr}$
ωl
ωr
srfl
ωR
δxtyrel
δxtyrer
srl
srr
μ$_{xl}$
μ$_{xr}$
Fx$_{groundl}$
Fx$_{groundr}$
ωR
−61
−62
−63
−64
−65
31
32
33
34
35
t
Adn
MosM