qmax Determination

Chris Rackauckas

April 28, 2019

```
using Distributed
addprocs()

@everywhere qs = 1.0 + 2.0.^(-5:2)

Error: On worker 2:
MethodError: no method matching +(::Float64, ::Array{Float64,1})
Closest candidates are:
  +(::Any, ::Any, !Matched::Any, !Matched::Any...) at operators.jl:502
  +(::Float64, !Matched::Float64) at float.jl:395
  +(::AbstractFloat, !Matched::Bool) at bool.jl:114
  ...
top-level scope at none:0
eval at ./boot.jl:328
#116 at /buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1
.1/Distributed/src/process_messages.jl:276
run_work_thunk at /buildworker/worker/package_linux64/build/usr/share/julia
/stdlib/v1.1/Distributed/src/process_messages.jl:56
run_work_thunk at /buildworker/worker/package_linux64/build/usr/share/julia
/stdlib/v1.1/Distributed/src/process_messages.jl:65
#102 at ./task.jl:259
#remotecall_wait#154(::Base.Iterators.Pairs{Union{},Union{},Tuple{},NamedTu
ple{(),Tuple{}}}, ::Function, ::Function, ::Distributed.Worker, ::Module, :
:Vararg{Any,N} where N) at /buildworker/worker/package_linux64/build/usr/sh
are/julia/stdlib/v1.1/Distributed/src/remotecall.jl:421
remotecall_wait(::Function, ::Distributed.Worker, ::Module, ::Vararg{Any,N}
 where N) at /buildworker/worker/package_linux64/build/usr/share/julia/stdl
ib/v1.1/Distributed/src/remotecall.jl:412
#remotecall_wait#157(::Base.Iterators.Pairs{Union{},Union{},Tuple{},NamedTu
ple{(),Tuple{}}}, ::Function, ::Function, ::Int64, ::Module, ::Vararg{Any,N
} where N) at /buildworker/worker/package_linux64/build/usr/share/julia/std
lib/v1.1/Distributed/src/remotecall.jl:433
remotecall_wait(::Function, ::Int64, ::Module, ::Vararg{Any,N} where N) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.1/Distr
ibuted/src/remotecall.jl:433
(::getfield(Distributed, Symbol("##163#165")){Module,Expr})() at ./task.jl:
259

...and 36 more exception(s).

times = Array{Float64}(undef,length(qs),4)

Error: UndefVarError: qs not defined

means = Array{Float64}(undef,length(qs),4)
```

```
Error: UndefVarError: qs not defined
```

```
@everywhere begin
  using StochasticDiffEq, DiffEqBase, DiffEqProblemLibrary, Random,
        Plots, ParallelDataTransfer, DiffEqMonteCarlo
  Random.seed!(99 + myid())
  full_prob = oval2ModelExample(largeFluctuations=true,useBigs=false)
  prob = remake(full_prob,tspan=(0.0,1.0))
  println("Solve once to compile.")
  sol = solve(prob,EM(),dt=1/2^(18))
  Int(sol.u[end][1]!=NaN)
  println("Compilation complete.")
  num_runs = 10000

  probs = Vector{SDEProblem}(undef,3)
  p1 = Vector{Any}(undef,3)
  p2 = Vector{Any}(undef,3)
  p3 = Vector{Any}(undef,3)
  ## Problem 1
  probs[1] = prob_sde_linear
  ## Problem 2
  probs[2] = prob_sde_wave
  ## Problem 3
  probs[3] = prob_sde_additive
end
```

```
Error: On worker 2:
ArgumentError: Package DiffEqBase not found in current path:
- Run `import Pkg; Pkg.add("DiffEqBase")` to install the DiffEqBase package
.

require at ./loading.jl:823
top-level scope at none:3
eval at ./boot.jl:328
#116 at /buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1
.1/Distributed/src/process_messages.jl:276
run_work_thunk at /buildworker/worker/package_linux64/build/usr/share/julia
/stdlib/v1.1/Distributed/src/process_messages.jl:56
run_work_thunk at /buildworker/worker/package_linux64/build/usr/share/julia
/stdlib/v1.1/Distributed/src/process_messages.jl:65
#102 at ./task.jl:259
#remotecall_wait#154(::Base.Iterators.Pairs{Union{},Union{},Tuple{},NamedTu
ple{(),Tuple{}}}, ::Function, ::Function, ::Distributed.Worker, ::Module, :
:Vararg{Any,N} where N) at /buildworker/worker/package_linux64/build/usr/sh
are/julia/stdlib/v1.1/Distributed/src/remotecall.jl:421
remotecall_wait(::Function, ::Distributed.Worker, ::Module, ::Vararg{Any,N}
 where N) at /buildworker/worker/package_linux64/build/usr/share/julia/stdl
ib/v1.1/Distributed/src/remotecall.jl:412
#remotecall_wait#157(::Base.Iterators.Pairs{Union{},Union{},Tuple{},NamedTu
ple{(),Tuple{}}}, ::Function, ::Function, ::Int64, ::Module, ::Vararg{Any,N
} where N) at /buildworker/worker/package_linux64/build/usr/share/julia/std
lib/v1.1/Distributed/src/remotecall.jl:433
remotecall_wait(::Function, ::Int64, ::Module, ::Vararg{Any,N} where N) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.1/Distr
ibuted/src/remotecall.jl:433
(::getfield(Distributed, Symbol("##163#165")){Module,Expr})() at ./task.jl:
259

...and 36 more exception(s).
```

```
println("Setup Complete")
```

```
Setup Complete
```

```
## Timing Runs

@everywhere function runAdaptive(i,k)
  sol =
    solve(prob,SRIW1(),dt=1/2^(8),abstol=2.0^(-15),reltol=2.0^(-10),maxIters=Int(1e12),qmax=qs[k])
  Int(any(isnan,sol[end]) || sol.t[end] != 1)
end

#Compile
monte_prob = MonteCarloProblem(probs[1])
```

```
Error: UndefVarError: probs not defined
```

```
test_mc =
    solve(monte_prob,SRIW1(),dt=1/2^(4),adaptive=true,num_monte=1000,abstol=2.0^(-1),reltol=0)
```

```
Error: UndefVarError: SRIW1 not defined
```

```
calculate_monte_errors(test_mc);
```

```
Error: UndefVarError: calculate_monte_errors not defined
```

0.1 qmax test on Oval2 Model

```
for k in eachindex(qs)
  ParallelDataTransfer.sendto(workers(), k=k)
  @everywhere Random.seed!(99 + myid())
  adaptiveTime = @elapsed numFails = sum(pmap((i)->runAdaptive(i,k),1:num_runs))
  println("k was $k.  The number of Adaptive Fails is $numFails.  Elapsed time was
    $adaptiveTime")
  times[k,4] = adaptiveTime
end
```

```
Error: UndefVarError: qs not defined
```

0.2 qmax test on other problems

```
for k in eachindex(probs)
  println("Problem $k")
  ## Setup
  prob = probs[k]
  ParallelDataTransfer.sendto(workers(), prob=prob)

  for i in eachindex(qs)
    ParallelDataTransfer.sendto(workers(), i=i)
    msim =
    solve(monte_prob,dt=1/2^(4),SRIW1(),adaptive=true,num_monte=num_runs,abstol=2.0^(-13),reltol=0,qma
    test_msim = calculate_monte_errors(msim)
    times[i,k] = test_msim.elapsedTime
    means[i,k] = test_msim.error_means[:final]
    println("for k=$kand i=$i, we get that the error was $(means[i,k])and it took
    $(times[i,k])seconds")
  end
end
```

```
Error: UndefVarError: probs not defined
```

```
using DiffEqBenchmarks
DiffEqBenchmarks.bench_footer(WEAVE_ARGS[:folder],WEAVE_ARGS[:file])
```

0.3 Appendix

These benchmarks are a part of the DiffEqBenchmarks.jl repository, found at: https://github.com/JuliaDi

To locally run this tutorial, do the following commands:

```
using DiffEqBenchmarks
DiffEqBenchmarks.weave_file("AdaptiveSDE","qmaxDetermination.jmd")
```

Computer Information:

```
Julia Version 1.1.0
Commit 80516ca202 (2019-01-21 21:24 UTC)
Platform Info:
  OS: Linux (x86_64-pc-linux-gnu)
  CPU: Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz
  WORD_SIZE: 64
  LIBM: libopenlibm
  LLVM: libLLVM-6.0.1 (ORCJIT, haswell)
```

Package Information:

```
Status: `/home/crackauckas/.julia/environments/v1.1/Project.toml`
[c52e3926-4ff0-5f6e-af25-54175e0327b1] Atom 0.8.5
[bcd4f6db-9728-5f36-b5f7-82caef46ccdb] DelayDiffEq 5.2.0
[bb2cbb15-79fc-5d1e-9bf1-8ae49c7c1650] DiffEqBenchmarks 0.1.0
[459566f4-90b8-5000-8ac3-15dfb0a30def] DiffEqCallbacks 2.5.2
[f3b72e0c-5b89-59e1-b016-84e28bfd966d] DiffEqDevTools 2.7.2+
[77a26b50-5914-5dd7-bc55-306e6241c503] DiffEqNoiseProcess 3.1.0
[055956cb-9e8b-5191-98cc-73ae4a59e68a] DiffEqPhysics 3.1.0
[a077e3f3-b75c-5d7f-a0c6-6bc4c8ec64a9] DiffEqProblemLibrary 4.1.0
[0c46a032-eb83-5123-abaf-570d42b7fbaa] DifferentialEquations 6.3.0
[b305315f-e792-5b7a-8f41-49f472929428] Elliptic 0.5.0
[e5e0dc1b-0480-54bc-9374-aad01c23163d] Juno 0.7.0
[7f56f5a3-f504-529b-bc02-0b1fe5e64312] LSODA 0.4.0
[c030b06c-0b6d-57c2-b091-7029874bd033] ODE 2.4.0
[54ca160b-1b9f-5127-a996-1867f4bc2a2c] ODEInterface 0.4.5
[09606e27-ecf5-54fc-bb29-004bd9f985bf] ODEInterfaceDiffEq 3.1.0
[1dea7af3-3e70-54e6-95c3-0bf5283fa5ed] OrdinaryDiffEq 5.5.0
[65888b18-ceab-5e60-b2b9-181511a3b968] ParameterizedFunctions 4.1.1
[91a5bcdd-55d7-5caf-9e0b-520d859cae80] Plots 0.24.0
[d330b81b-6aea-500a-939a-2ce795aea3ee] PyPlot 2.8.1
[90137ffa-7385-5640-81b9-e52037218182] StaticArrays 0.10.3
```

```
[789caeaf-c7a9-5a7d-9973-96adeb23e2a0] StochasticDiffEq 6.1.1
[c3572dad-4567-51f8-b174-8c6c989267f4] Sundials 3.3.0+
[92b13dbe-c966-51a2-8445-caca9f8a7d42] TaylorIntegration 0.4.1
[44d3d7a6-8a23-5bf8-98c5-b353f8df5ec9] Weave 0.9.0
```