

aprof: Amdahl's profiler, directed optimization made easy

Marco D. Visser

October 31, 2013

1 Introduction

Amdahl's profiler or *aprof* is a R package meant to help to evaluate whether and where to focus code optimization using Amdahl's law and visual aids based on line profiling. The package *aprof* is an addition to R's standard profiling tools and is not a wrapper for them. Amdahl's profiler organises profiling output files (including memory profiling, in a next release) in a visually appealing way and helps to identify the most promising sections of code to optimize. It is meant to help to balance development vs. execution time.

Here is an example of some basic *aprof* operations, showing the difference in system resources usage between pre-allocating objects in the memory of growing them

```
> require(aprof)
> # create function to profile
>   foo <- function(N){
+     preallocate<-numeric(N)
+     grow<-NULL
+     for(i in 1:N){
+       preallocate[i]<-N/(i+1)
+       grow<-c(grow,N/(i+1))
+     }
+   }
>   #save function to a source file and reload
>   dump("foo",file="foo.R")
>   source("foo.R")
>   # Profile the function
>   Rprof("foo.out",line.profiling=TRUE)
>   foo(5e4)
>   Rprof(append=FALSE)
>   # Create a aprof object
>   fooaprof<-aprof("foo.R","foo.out")
>   fooaprof
```

Source file:
foo.R (9 lines).

Call Density and Execution time per line number:

Line	Call Density	Time Density (s)
Totals:		
Calls	1	
Time (s)	0	(interval = 0.02 (s))

The table above gives some basic output, and shows the lines that were sampled by the profiler, and the amount of time spent in each line. In this simple example it's clear that line 7, takes the most time and should be the target of optimization. This is also shown in the next figure:

```
> plot(fooaprof)
```