

Lecture 9: Writing Fast Matlab Code

Cinna Wu

cinnowu@math.berkeley.edu

Department of Mathematics
University of California, Berkeley

Math 98 Introduction to MATLAB programming

Announcements

- **HW4** is due this Thursday!

The main three

- **Comment:** Optimized codes tend to be harder to read. Remember to comment.
- **Only optimize when it is time:** Decide if it is worth it before optimizing. Don't do it until you have a working code.
- **Only optimize where needed** Make sure you are optimizing something that actually needs to be optimized. Check it is a bottleneck first.

Tic Toc

Measures time in matlab. Look up in help menu.

Exercise

Use tic toc to measure the time it takes for Newton's algorithm to converge. Compare the speed with computing `sqrt(5)` in matlab.

Profiler

Helps to find the bottlenecks in a program. Use **profile on** to turn the profiler on and **profile viewer** to view the profile.

Example

```
function result = example1(Count)
for k = 1:Count
result(k) = sin(k/50);
if result(k) < -0.9
result(k) = gammaln(k);
end
end
Use profile to analyze example1(5000).
```

Array Preallocation

Matlab dynamically augments rows and columns

```
a = 2; a(3,3) = 3;
```

Avoid this when possible by preallocating when possible.

Compare the following scripts:

```
a(1) = 1; b(1) = 0;
```

```
for k = 2:8000
```

```
a(k) = 0.99803 * a(k - 1) - 0.06279 * b(k - 1);
```

```
b(k) = 0.06279 * a(k - 1) + 0.99803 * b(k - 1);
```

```
end
```

```
a = zeros(1,8000); b = zeros(1,8000);
```

```
a(1) = 1; b(1) = 0;
```

```
for k = 2:8000
```

```
a(k) = 0.99803 * a(k - 1) - 0.06279 * b(k - 1);
```

```
b(k) = 0.06279 * a(k - 1) + 0.99803 * b(k - 1);
```

```
end
```

Use Vectorized Functions when Possible

Instead of this:

```
function d = minDistance(x,y,z)
nPoints = length(x);
d = zeros(nPoints,1);
for k = 1:nPoints
d(k) = sqrt(x(k)^2 + y(k)^2 + z(k)^2);
end
d = min(d);
```

Do this:

```
function d = minDistance(x,y,z)
d = sqrt(x.^2 + y.^2 + z.^2);
d = min(d);
```

Other Vectorized Functions

Common Built-in Function

min, max, repmat, meshgrid, sum,
cumsum, diff, prod, cumprod, filter
find, any, all

Example

Given a vector x , remove all the negative entries from the vector.

Exercise 1

Find the bottlenecks of the following code using tic toc or profiler and fix them.

```
a = 0;
count = 0;
for k = 1:10000
    v = exp(rand(1)*rand(1));
    if v > 0.5
        count = count + 1;
        a(count) = v;
    end
end
a = a(1:count);
```