MDO via MDF

20/20

Doug Shi-Dong 260466662
MECH-579 Multidisciplinary Design Optimization
Department of Mechanical Engineering, McGill University

November 29, 2013

1 Introduction

Through a multidisciplinary feasible design (MDF) framework, find the minimum of $f(x, y)$ dictated by R_1 and R_2.

$$\begin{aligned} \text{minimize} \quad & f(x,y) = -20e^{-[(x_1-1)^2+0.25(x_2-1)^2]} + y_1 + cos(y_2) \\ \text{with respect to} \quad & x_1, x_2 \; \epsilon \; \mathbb{R}^n \\ \text{where} \quad & R_1(x, y_1(x, y_2)) : y_1 = -3e^{-[(x_1+1)^2+0.25(x_2+1)^2]} + sin(y_2) \\ & R_1(x, y_2(x, y_1)) : y_2 = -3e^{-[5(x_1-3)^2+0.25(x_2-3)^2]} + e^{-y_1} \end{aligned}$$

The report contains the analysis of the derivatives of the objective function and the use of Quasi-Newton to find the optimum solution.

2 Derivatives

The derivative is computed through first-order finite difference (FD1), second-order finite difference (FD2), complex-step (CS), direct method (DM) and adjoint method (AM). It is taken at point $(1, 1)$.

good.

Figure 1 shows the difference between FD1, FD2 and CS in comparison to the DM. The DM and AM are exactly the same down to 10E-16.

The FD1's error decreases at order 1 and the FD2 and CS's error decreases at order 2, as expected from first and second order methods. For FD, the round-off error starts taking over and increases the total error. The CS does not suffer from that since there is no substraction of two similar numbers and therefore the error goes down to machine zero as the step is decreased.

This plot is not only useful to compare those three methods but also confirms the accuracy of the DM and AM.

Note: the derivations for the DM and AM are appended at the end of the report.

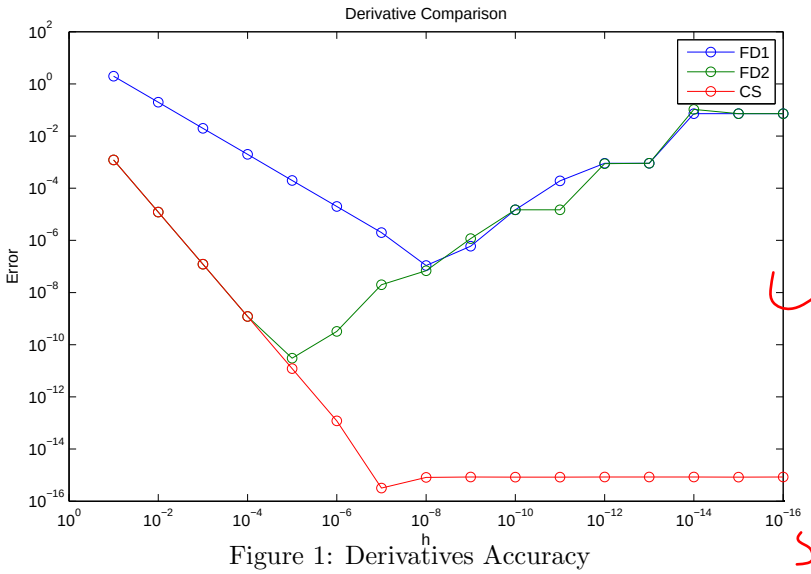

Figure 1: Derivatives Accuracy

State the exact direct and adjoint gradient values.

3 Optimization

The Quasi-Newton method is used for the line-search algorithm. A backtracking algorithm determines the size of α. For a fixed step length, the first iteration should be large enough to get out of the noisy region and the next few iterations should be small in order to condition the pseudo-Hessian. The convergence is determined when $\|\nabla\mathcal{L}\| < 10\text{E-}12$. For the starting point (-0.1,-1), the minimum point is found at (0.998186267926089, 0.998186267761909).

A very important feature that is not shown on those plots is the number of iterations required to calculate y. The convergence of y is determined when the y's found satisfy the governing equations at a precision of 10E-15. It requires between 40 to 50 iterations of solving each governing equations in order to find y for a given x. For every optimization step, y is required at x. In the case of a backtracking algorithm, $f(x, y)$ is calculated many time for different x's. In this assignment around 5000 iterations between y's are required, for only 14 optimization steps. The CPU time is next to nothing, but for a real problem, iterating between y's is very costly.

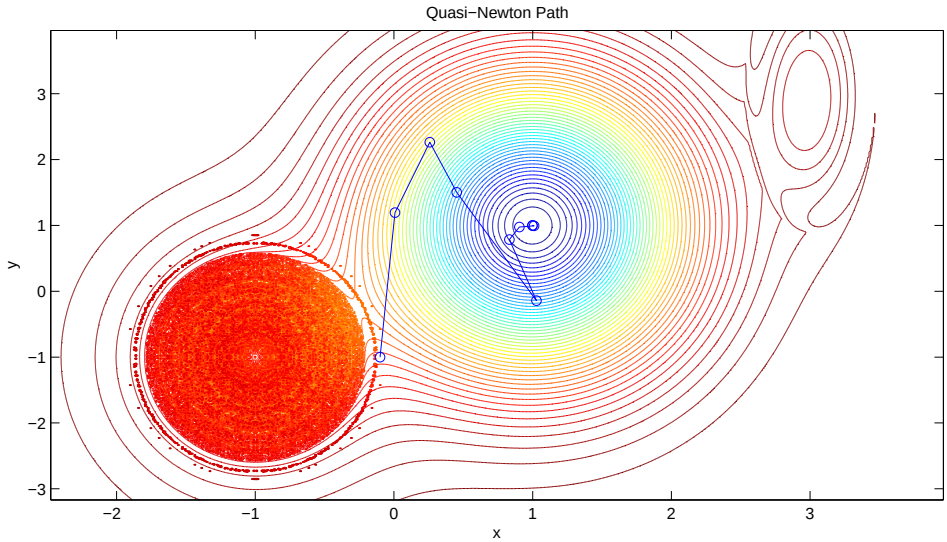

Figure 2: Optimization Path, variable α

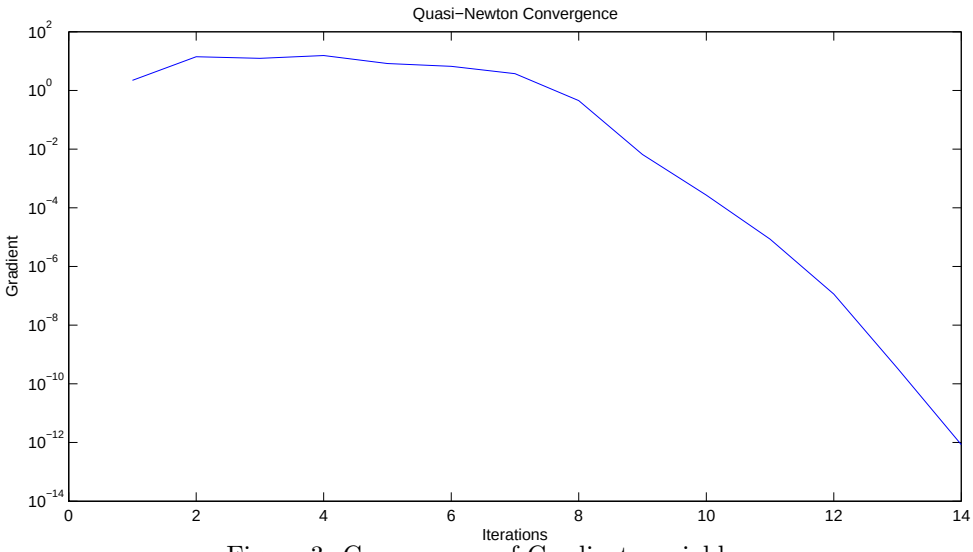

Figure 3: Convergence of Gradient, variable α

4 Conclusion

The FD methods decrease in error with the step, but starts increasing when the step becomes small enough for the round-off error to dominate. The CS does not have any round-off error. Those derivatives also confirm the exactitude of the DM and AM.

The MDF framework is very expensive because of the necessity to iterate between y's.

5 Derivations

The following derivations have been done on Maple:
Definition of objective function f and governing equations R1 and R2

$$f := -20\,\mathrm{e}^{-(x_1-1)^2-0.25\,(x_2-1)^2} + y_1 + \cos(y_2)$$

$$-20\,\mathrm{e}^{-(x_1-1)^2-0.25\,(x_2-1)^2} + y_1 + \cos(y_2)$$

$$R_1 := -3\,\mathrm{e}^{-(x_1+1)^2-0.25\,(x_2+1)^2} + \sin(y_2) - y_1$$

$$-3\,\mathrm{e}^{-(x_1+1)^2-0.25\,(x_2+1)^2} + \sin(y_2) - y_1$$

$$R_2 := -3\,\mathrm{e}^{-5\,(x_1-3)^2-0.25\,(x_2-3)^2} + \mathrm{e}^{-y_1} - y_2$$

$$-3\,\mathrm{e}^{-5\,(x_1-3)^2-0.25\,(x_2-3)^2} + \mathrm{e}^{-y_1} - y_2$$

Partial derivatives of the f

$$pfpx1 := \frac{d}{dx_1} f$$

$$-20\,(-2\,x_1 + 2)\,\mathrm{e}^{-(x_1-1)^2-0.25\,(x_2-1)^2}$$

$$pfpx2 := \frac{d}{dx_2} f$$

$$-20\,(-0.50\,x_2 + 0.50)\,\mathrm{e}^{-(x_1-1)^2-0.25\,(x_2-1)^2}$$

$$pfpy1 := \frac{d}{dy_1} f$$

$$1$$

$$pfpy2 := \frac{d}{dy_2} f$$

$$-\sin(y_2)$$

Partial derivatives of governing equations f

$$pR1px1 := \frac{d}{dx_1} R_1$$

$$-3\,(-2\,x_1 - 2)\,\mathrm{e}^{-(x_1+1)^2-0.25\,(x_2+1)^2}$$

$$pR1px2 := \frac{d}{dx_2} R_1$$

$$-3\,(-0.50\,x_2 - 0.50)\,\mathrm{e}^{-(x_1+1)^2-0.25\,(x_2+1)^2}$$

$$pR1py1 := \frac{d}{dy_1} R_1$$

$$-1$$

$$pR1py2 := \frac{d}{dy_2} R_1$$

$$\cos(y_2)$$

$$pR2px1 := \frac{d}{dx_1} R_2$$

$$-3\,(-10\,x_1 + 30)\,\mathrm{e}^{-5\,(x_1-3)^2-0.25\,(x_2-3)^2}$$

$$pR2px2 := \frac{d}{dx_2} R_2$$

$$-3\,(-0.50\,x_2 + 1.50)\,\mathrm{e}^{-5\,(x_1-3)^2-0.25\,(x_2-3)^2}$$

$$pR2py1 := \frac{d}{dy_1} R_2$$

$$-e^{-y_1}$$

$$pR2py2 := \frac{d}{dy_2} R_2$$

$$-1$$

The following statements are in MATLAB to compute the derivatives:

Direct Method:

```
A=[pR1py1 pR1py2; pR2py1 pR2py2];
Ainv=inv(A);

dydx(:,1)=-Ainv*[pR1px(1);pR2px(1)];
dydx(:,2)=-Ainv*[pR1px(2);pR2px(2)];

dfdx(1)=pfpx(1)+pfpy1*dydx(1,1)+pfpy2*dydx(2,1);
dfdx(2)=pfpx(2)+pfpy1*dydx(1,2)+pfpy2*dydx(2,2);

dfdx=dfdx';
```

Adjoint Method:

```
A=[pR1py1 pR2py1;pR1py2 pR2py2];
Ainv=inv(A);

psi=-Ainv*[pfpy1;pfpy2];

dfdx(1)=pfpx(1)+psi(1)*pR1px(1)+psi(2)*pR2px(1);
dfdx(2)=pfpx(2)+psi(1)*pR1px(2)+psi(2)*pR2px(2);

dfdx=dfdx';
```