Homework No 12 (Conjugate Gradient Method)

Consider the Conjugate gradient method for unconstrained minimization of large scale problems :

$$\min_{x \in \Re^n} f(x_1, x_2, ..., x_n) :$$

Please use the following software (or write your own code) to implement the Fletcher Reeves and Polak Ribiere as well as the positive Polak Ribiere updates for the Conjugate gradient method:

1. Use the file cg.tar to make a directory CG+ and test F-R, P-R and positive P-R update methods for the conjugate gradient algorithm. Use the Woods function described below as well as its gradient.

2. Use the Recipes code frprmn.f implementing C-G in either F-R or P-R updates versions but with different step size search for testing same function.

3. Test the conminf.f code with nmeth=0 in which case it implements an advanced C-G method.(see Shanno and Phua reading material)

4. Use the UMCGG/ (Single precision) from the IMSL library as advised in class by testing it on hilbert.math.fsu.edu. It minimizes a function of N variables using a conjugate gradient algorithm and with a user-supplied gradient. (Based on Powell 1977).

Please modify the main driver routine or write equivalent code by yourself and test it for the difficult 4 variables Woods function:

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2$$
$$+10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1)$$

And its gradient.

a) Test the code using the starting point:

$x = (-3, -1, -3, -1)^T$

Use uniform accuracy criterion chosen by you.

b) Compare the performance of the various CG routines in terms of CPU time, number of iterations and gradient norm for all the above cases.
c) Find out which routine from either CG+, (3 possibilities), frprmn.f, conminf.f or UMCGG is the most performing in terms of number of iterations, function calls, final value of function and gradient norm.

d) Please discuss and provide explanations for the results obtained.