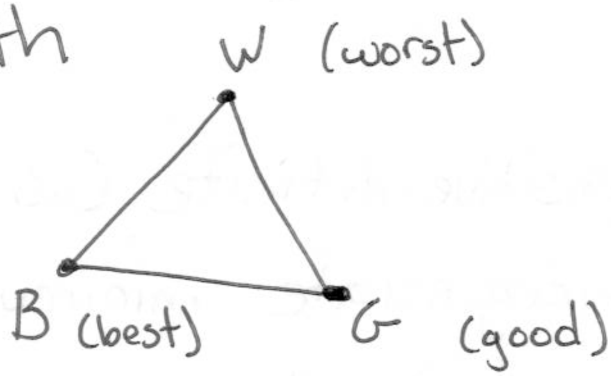
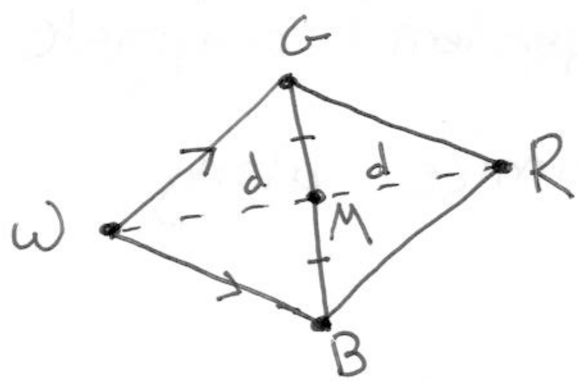
Nelder-Mead Algorithm

The basic idea of Nelder-Mead is that you will plan to start with



so that $f(B) \leq f(G) \leq f(W)$. We want to replace W with another vertex. We do this with a set of possible moves.

First option: reflection in BG



Since f goes down from W to G and from W to B, we guess that f might be smaller on the other side of BG.

We test this by constructing ~~the~~ M, the midpoint of BG and extending WM past M to R so that $WM = MR$.

We then evaluate at R.

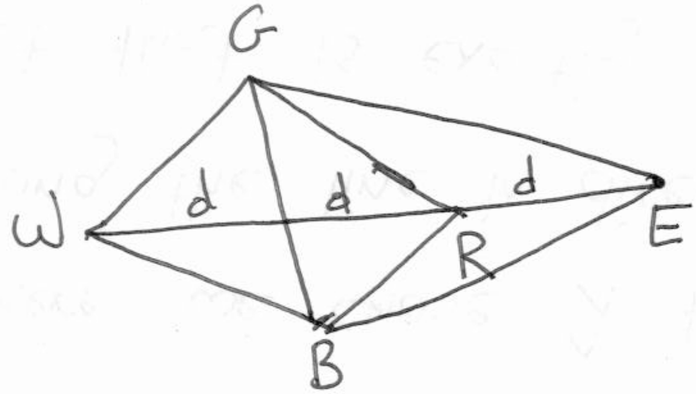
Case 1. $f(R) < f(G)$

We have improved things by going to R, so our next triangle will go in this direction.

If $f(B) < f(R)$, then we should

probably turn towards the BR edge. We replace W by R and stop.

If $f(R) < f(B)$, we are really winning with this direction. We try to "double down" by constructing



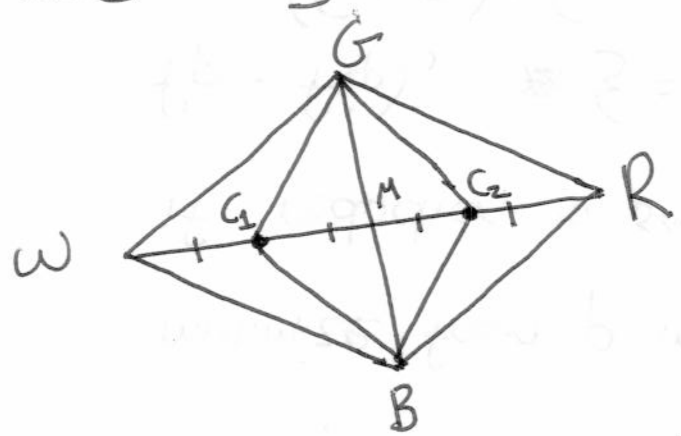
the "extended" point E.

If $f(E) < f(B)$
    replace W with E
Else
    replace W with R

Case 2. $f(R) \geq f(G)$.

We first check to see if $f(R) < f(W)$. If so, we have made progress, and we replace ~~R~~ W with R and stop.

If $f(R) \geq f(W)$, we need an entirely new move type. It seems that BG may lie in a sort of valley, so we try



C_1 = midpoint of WM

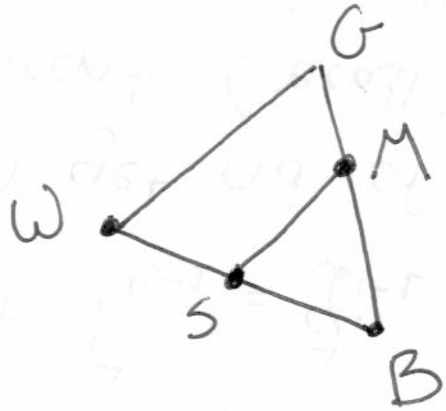
C_2 = midpoint of MR

We let C = whichever one of C_1 and C_2 has the lowest function value. This is called "contraction".

If $f(c) < f(W)$,

replace W with C and stop.

Otherwise, we'll have to introduce a last ditch move, called "shrink".



M = midpoint of BG

S = midpoint of WB

Replace $\{W, G\}$ with $\{S, M\}$ (the ~~{depend~~ order depends on whether $f(S) < f(M)$ or vice versa).

The generalization to higher dimensional problems will follow.