Pseudo-code of current version of my algorithm:

~~read boundaires of variables;~~

initialize starting point $x^0 \in \mathbb{R}^n$;

initialize radius $r$ and size of sampling points *num*;

initialize a blank list to store optimal values

**repeat**

  shuffle the order of variables;

  **for** index $i_1$ to $i_k, k \in \{1,2,\dots,n\}$:

    **repeat**

      sampling along single direction within range $[x_{i_k} - r, x_{i_k} + r]$;

      call Alamopy to get the surrogate model based on sampling data points;

      call Baron to get the optimal point of surrogate model;

      **if** ~~the difference between actual black-box value and calculation result of optimal point is big;~~

        decrease $r$; increase *num*;

      **else**:

        increase $r$; decrease *num*;

    **until** the actual black-box value of optimal point $\approx$ minimal value obtained by Baron;

    update $x_{ik}$ with new optimal point; and append optimal value to solution list;

**until** termination is satisfied