Dichotomous search

Let us consider a functional $\mathcal{J}(\boldsymbol{X}) : \mathbb{R}^N \mapsto \mathbb{R}$, defined as

$$\mathcal{J}(\boldsymbol{X}) = -\frac{N+1}{2}\boldsymbol{X} \cdot \left[\underline{\mathcal{C}}\boldsymbol{X}\right] - \boldsymbol{B}_0 \cdot \boldsymbol{X} \quad \text{where } \boldsymbol{B}_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ (N+1) \end{pmatrix}, \quad \underline{\mathcal{C}} = \begin{pmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & -2 & 1 \\ 0 & \cdots & 0 & 1 & -2 \end{pmatrix}$$

$\boldsymbol{B}_0$ is a vector of size N and $\underline{\mathcal{C}}$ is a $N \times N$ matrix. Then, for a given vector $\boldsymbol{u}$, we define functionals $f_k^{\boldsymbol{u}}(\rho) : \mathbb{R} \mapsto \mathbb{R}$ as :

$$f_k^{\boldsymbol{u}}(\rho) = \mathcal{J}(\boldsymbol{u} + \rho \boldsymbol{e}_k) \quad \text{where} \quad \boldsymbol{e}_{k=1,\cdots,N}, \text{ define the standard basis of } \mathbb{R}^N$$

The aim here is to solve the minimization of $\mathcal{J}(\boldsymbol{X})$ by computing minimum of $f_k^{\boldsymbol{u}}(\rho)$ on intervals $[a, b]$.

Dichotomous search algorithm to minimize $f_k(\rho)$

1. Choose `VecU`, `k`, a tolerance `Myeps` and a maximum of iterations `idmax`.
2. Initialized `Mya` and `Myb` with `Myb > Mya ,`
3. $f_k^{\boldsymbol{u}}(\rho) \equiv$ `MyF(VecU, k, Rho)` a fortran, C or silab function

```
DO id = 1, idmax
  Rho1  = ( Mya + Myb - Myeps)/2.0
  Rho2  = ( Mya + Myb + Myeps)/2.0
  IF( MyF(MyVecU, Myk, rho1)  <= MyF(MyVecU, Myk, rho2) ) THEN
     Myb = rho2
  ELSE
     Mya = Rho1
  END IF

  IF( ABS(Myb-Mya) < 2*eps ) Exit

END DO
```

4. Approximated minimum `Xs = (Mya + Myb)/2.0`

Application : Relaxation Method

1. For $N = 2$
   - verify that the minimum of $\mathcal{J}(\boldsymbol{X})$ is obtained with $\boldsymbol{X}_* = \left(\frac{1}{3},\ \frac{2}{3}\right)^T$
   - verify that, for $U = 0$, $f_1^{\mathbf{0}}(\rho) = 3\rho^2$ and $f_2^{\mathbf{0}}(\rho) = 3\rho^2 - 3\rho$
   - using `eps=0.01` and `VecU=(/0, 0/)` compute the approximated minimum `Xs` of `MyF(VecU, 1, Rho)` with the initialization `a = -0.5` and `b = 0.5`.
   - using `eps=0.01` and `VecU= VecU + (/Xs, 0/)` compute the approximated minimum `Ys` of `MyF(VecU, 2, Rho)` with the initialization `a = -0.5` and `b = 0.5`.
   - using `eps=0.01` and `VecU= VecU + (/0, Ys/)` compute the approximated minimum `Xs` of `MyF(VecU, 1, Rho)` with the initialization `a = -0.5` and `b = 0.5`.
   - using `eps=0.01` and `VecU= VecU + (/Xs, 0/)` compute the approximated minimum `Ys` of `MyF(VecU, 2, Rho)` with the initialization `a = -0.5` and `b = 0.5`.
   - Compare the final `VecU= VecU + (/0, Ys/)` with $\boldsymbol{X}_*$
2. For $N = 49$, can you find the exact minimum $\boldsymbol{X}_*$? The approximated minimum is computed as :

```
VecU  = 0
eps   = 1.0E-9

DO i = 1, 100*N
   DO k = 1, N

      a= -0.5 ; b = 0.5
      VecU(k) = VecU(k) + Dichotomous(eps, k, VecU, a, b, 200*N)
   END DO
     IF(MOD(i,10) == 0)  Write(6,*) i, " Xs(1:2) = " , VecU(1:2)
END DO
```

   where `Dichotomous(eps, k, VecU, a, b, 200*N)` is a function computing the minimum of $f_k^{\boldsymbol{u}}(\rho)$. The final `VecU` is an approximated minimum. Follow the convergence of this relaxation.