$\text{def } bisectionEXP(left, right):$

$\quad P_L, P_R = P_{opt}(left), P_{opt}(right)$

$\quad if \ (left \leq right) \ and \ (P_L \ != P_R):$

$$mid = \frac{left + right}{2}$$

$\quad\quad P_M = P_{opt}(mid)$

$\quad\quad if \ (P_L != P_M):$

$\quad\quad\quad\quad bisectionEXP(left, \ mid)$

$\quad\quad if \ (P_R != P_M):$

$\quad\quad\quad\quad bisectionEXP(mid, \ right)$

```
def AdaNEXUS(CC_i) :
    S_now, P_now = InitializeSeed(CC_i)
    C_now = Cost(P_opt, S_now)
    step, Δ_now = 1 , [−1,0]
    while (There exist next point):
        S_proxy = S_now + step * Δ_now
        C_proxy, P_proxy = Cost(P_opt , S_proxy)
        S_next , Δ_next = Correct(CC_i, C_proxy)
        C_next, P_next = Cost(P_opt , S_next)
        if (max(C_next/CC_i , CC_i/C_next) ≤ (1 + α)) :
            Δ_now = TuneDir(Δ_now , Δ_next)
            bisectionEXP(S_now , S_next)
            S_now , step = S_next , 2 * step
        else :
            if step > 1 :
            step = step / 2
        else :
            Δ_now = Rotate(S_now , Δ_now)
```