Se presenta el algoritmo general.

Algorithm 1 Iterated Greedy

1: **procedure** $Iterated\,Greedy$
2: $s_0 = GeneralInitialSolution$
3: **repeat**
4: $s_p = \text{Destruction}(s^*)$
5: $s' = \text{Construction}(s_p)$
6: $s^* = \text{AcceptanceCriterion}(s^*, s')$

Algoritmo para la solución inicial.

Algorithm 2 General Inicial Solution

1: Algorithm local search for vector x.
2: Algorithm minimum weight and maximum distance search for vector z.

Algoritmo para la destrucción.

Algorithm 3 Destruction

1: Determinate s_1, s_2. ▷ s_1 and s_2 is a vector
2: **if** long $s_1 > m$ or long $s_2 > n$ **then**
3: **repeat** step 1.
4: Eliminate z_{s_1} from vector z.
5: Eliminate x_{s_2} from vector x.

Algoritmo para la contrucción.

Algorithm 4 Construction

1: $s_{p_1} = (z_1, z_2, \cdots, z_{m-s_1})$, $s_{p_2} = (x_1, x, \cdots, x_{n-s_2})$
2: $s_1^{'}$ = Algoritmo en base al cociente de rendimiento $r_i = \dfrac{b_i}{v_i}$, $\forall s_1$
3: $s_2^{'}$ = Algoritmo del vecino más cercano $\forall p_2$.

Algoritmo para la aceptación.

Algorithm 5 Acceptance Criterion

1: Acceptance Criterion
2: **if** $G(s_2^{'}, s_1^{'}) > G(s_2^*, s_1^*)$ **then**
3: $s_2^* \longleftarrow s_2^{'}, s_1^* \longleftarrow s_1^{'}$