

procedure MAP-ELITES ALGORITHM (SIMPLE, DEFAULT VERSION)

$(\mathcal{P} \leftarrow \emptyset, \mathcal{X} \leftarrow \emptyset)$

▷ Create an empty, N -dimensional map of elites: {solutions \mathcal{X} and their performances \mathcal{P} }

for iter = 1 \rightarrow I **do**

▷ Repeat for I iterations.

if iter < G **then**

▷ Initialize by generating G random solutions

$\mathbf{x}' \leftarrow \text{random_solution}()$

else

▷ All subsequent solutions are generated from elites in the map

$\mathbf{x} \leftarrow \text{random_selection}(\mathcal{X})$

▷ Randomly select an elite x from the map \mathcal{X}

$\mathbf{x}' \leftarrow \text{random_variation}(\mathbf{x})$

▷ Create x' , a randomly modified copy of x (via mutation and/or crossover)

$\mathbf{b}' \leftarrow \text{feature_descriptor}(\mathbf{x}')$

▷ Simulate the candidate solution x' and record its feature descriptor \mathbf{b}'

$p' \leftarrow \text{performance}(\mathbf{x}')$

▷ Record the performance p' of x'

if $\mathcal{P}(\mathbf{b}') = \emptyset$ or $\mathcal{P}(\mathbf{b}') < p'$ **then**

▷ If the appropriate cell is empty or its occupants's performance is $\leq p'$, then

$\mathcal{P}(\mathbf{b}') \leftarrow p'$

▷ store the performance of x' in the map of elites according to its feature descriptor \mathbf{b}'

$\mathcal{X}(\mathbf{b}') \leftarrow \mathbf{x}'$

▷ store the solution x' in the map of elites according to its feature descriptor \mathbf{b}'

return feature-performance map (\mathcal{P} and \mathcal{X})