Which point in the convex polygon is closest to the goal?

The presented method finds among the points that belong to a convex polygon in the Euclidean plane the one point with minimum distance to a goal point.

Algorithm 1 Distance Minimization Incrementation

Input: halfplane H, goal point p°, prior solution $\tilde{p}^*$, prior halfplanes $\left\{\tilde{H}_j\right\}_{j=1}^{m}$

Output: solution point p^* being closest to p° while belonging to $\bigcap_{j=1}^{m} \tilde{H}_j \cap H$

$p^* \leftarrow \tilde{p}^*$ *//initialize the solution as the prior solution*
if $p^* \notin H$ **then**
  $p^* \leftarrow$ projectPointOrthogonallyOnHalfplaneBoundary(p°, H)
  $p_1 \leftarrow p_1^{H,\infty}$ *//point at infinity at one end of H's boundary*
  $p_2 \leftarrow p_2^{H,\infty}$ *//point at infinity at the other end of H's boundary*
  for $j \leftarrow 1, ..., m$ **do**
    if $(p_1 \notin \tilde{H}_j) \,\&\, (p_2 \in \tilde{H}_j)$ **then**
      $p_1 \leftarrow$ intersectHalfplaneBoundaries$(\tilde{H}_j, H)$ *//update segment* $[p_1, p_2]$
    else if $(p_1 \in \tilde{H}_j) \,\&\, (p_2 \notin \tilde{H}_j)$ **then**
      $p_2 \leftarrow$ intersectHalfplaneBoundaries$(\tilde{H}_j, H)$ *//update segment* $[p_1, p_2]$
    else if $(p_1 \notin \tilde{H}_j) \,\&\, (p_2 \notin \tilde{H}_j)$ **then**
      except "infeasible" *//terminate;* p_1*,* p_2 *serve only to determine this*
    end if *//The above conditionals maintain the feasible boundary segment*
    if $p^* \notin \tilde{H}_j$ **then**
      $p^* \leftarrow$ intersectHalfplaneBoundaries$(\tilde{H}_j, H)$ *//update the solution*
    end if
  end for
end if
return p^*

Algorithm 2 Incremental Distance Minimization

Input: goal point p°, halfplanes $\{H_i\}_{i=1}^{n}$

Output: solution point p^* being closest to p° while belonging to $\bigcap_{i=1}^{n} H_i$

$p^* \leftarrow p^\circ$
for $i \leftarrow 1, ..., n$ **do**
  $p^* \leftarrow$ distanceMinimizationIncrementation $\left(H_i, p^\circ, p^*, \{H_j\}_{j=1}^{i-1}\right)$
end for
return p^*

Being very similar to previous work [1], the method sligthly varies the approach of incremental linear programming [2], [3]. Algorithm 1 receives the convex polygon as a set of halfplanes whose intersection defines the polygon. It requires to know the solution of the same problem with one halfplane less and which is the respective halfplane. The algorithm allows to incrementally build the final solution for any convex polygon, by solving first the problem without any halfplanes, then adding one more halfplane and solving again, and so on, until it has processed all the halfplanes. Algorithm 2 implements this approach.

References

[1] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics research*, pp. 3–19, Springer, 2011.

[2] R. Seidel, "Small-dimensional linear programming and convex hulls made easy," *Discrete & Computational Geometry*, vol. 6, no. 3, pp. 423–434, 1991.

[3] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational geometry: algorithms and applications.* Springer, 2008.