Polygon-Constrained Motion Planning Problems

Davide Bilò[1], Yann Disser[2], Luciano Gualà[3], Matúš Mihal'ák[4], Guido Proietti[5,6(✉)], and Peter Widmayer[4]

[1] Dipartimento di Scienze Umanistiche e Sociali, Università di Sassari, Sassari, Italy
[2] Institut für Mathematik, Technische Universität Berlin, Berlin, Germany
[3] Dipartimento di Ingegneria dell'Impresa, Università di Roma Tor Vergata, Rome, Italy
[4] Institut für Theoretische Informatik, ETH, Zürich, Switzerland
[5] Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica, Università dell'Aquila, Coppito L'Aquila, Italy
[6] Istituto di Analisi dei Sistemi ed Informatica, CNR, Roma, Italy
guido.proietti@univaq.it

Abstract. We consider the following class of polygon-constrained motion planning problems: Given a set of k centrally controlled mobile agents (say *pebbles*) initially sitting on the vertices of an n-vertex simple polygon P, we study how to plan their vertex-to-vertex motion in order to reach with a minimum (either *maximum* or *total*) movement (either in terms of *number of hops* or *Euclidean distance*) a final placement enjoying a given requirement. In particular, we focus on final configurations aiming at establishing some sort of *visual connectivity* among the pebbles, which in turn allows for wireless and optical intercommunication. Therefore, after analyzing the notable (and computationally tractable) case of gathering the pebbles at a *single* vertex (i.e., the so-called *rendezvous*), we face the problems induced by the requirement that pebbles have eventually to be placed at: (i) a set of vertices that form a *connected* subgraph of the *visibility graph* induced by P, say $G(P)$ (*connectivity*), and (ii) a set of vertices that form a *clique* of $G(P)$ (*clique-connectivity*). We will show that these two problems are actually hard to approximate, even for the seemingly simpler case in which the hop distance is considered.

1 Introduction

In many practical applications a number of centrally controlled devices need to be moved from an initial positioning towards a final configuration so that a desired task can be completed. In particular, in settings like robotics and sensor networking, the devices generally happen to have a limited transmission and reception capability, and thus to establish some kind of reciprocal communication they need

This work was partially supported by the Research Grant PRIN 2010 "ARS TechnoMedia", funded by the Italian Ministry of Education, University, and Research. Part of this work was developed while Guido Proietti was visiting ETH.

P. Flocchini et al. (Eds.): ALGOSENSORS 2013, LNCS 8243, pp. 67–82, 2014.
DOI: 10.1007/978-3-642-45346-5_6,

to build an obstacle-free *ad-hoc network*. However, by any respects, movements are expensive, and so this repositioning procedure should be accomplished in such a way that some distance-related objective function is minimized.

In this paper, we assume the underlying environment is a *simple polygon*, say P, and the moving devices (*pebbles*, in the sequel) are initially placed on vertices of P. In our setting, pebbles can only move within the polygon through a *vertex-to-vertex polygonal path*, and so they will reach a final position which coincides with a polygon vertex. This restriction about the initial, intermediate, and ending position of the pebbles is motivated by the fact that vertices are a notable position in a polygon, for which several well-studied classes of computational geometry problems (e.g., art-gallery guarding, facility location, etc.) have been considered. Moreover, from a more practical point of view, we point out that recently there has been a growing attention towards *limited-sensing* robotic devices, which are built in such a way that they are able to only detect very minimal information about the surrounding environment. In particular, the so-called *combinatorial robots* [12] are only able to move to visible corners of the (planar) region they are embedded in, i.e., the vertices of a polygon. Therefore, we study a set of *motion planning* (i.e., centrally managed) problems that arise by the combination of three different final positioning goals and a pair of movement optimization functions, which will be computed with respect to two different distance concepts. More precisely, we first focus our study on a scenario where we want the pebbles to be moved to a *single vertex* (RV, which stands for *rendez-vous*) of P. In fact, gathering at a single vertex will enable pebbles to exchange information in a setting where long-range communication is not allowed. Then, we turn our attention to the more general case in which pebbles have to form a *connected subgraph* (Con) of the *visibility graph* of P. Recall that such a graph has a node for each polygon vertex, and an edge for each pair of polygon vertices which can be joined by a straight line contained in the interior or the boundary of polygon P. Thus, quite naturally, we focus on the visibility graph of P, since intervisibility between polygon vertices turns out to enable wireless or optical connection among devices. Finally, in order to consider the plausible case in which a mutual direct connection among pebbles is needed, we analyze the problem in which they have to form a *clique* (Clique) in the visibility graph. For all these problems, we consider both the minimization of the *overall movement* (Sum) and the *maximum movement* (Max) of the pebbles. To this respect, these functions will be measured both in terms of the classic *Euclidean distance* (ED) covered by the pebbles, and with regard to the *hop distance* (HD) measure, i.e., that in which the distance between two vertices in P is given by the minimum number of edges in any vertex-to-vertex polygonal path in P connecting the two vertices. This latter type of distance is important in many practical cases since it resorts to the number of turns that a device must take all along the way.

Related Work. Although movement problems were deeply investigated in a distributed setting (see [11] for a survey), quite surprisingly the centralized counterpart has received attention from the scientific community only very recently.

The first paper which defines and studies these problems in this latter setting is [6]. In their work, the authors study the problem of moving the pebbles on a graph G of n vertices so that their final positions form any of the following configurations: connected component, path (directed or undirected) between two specified nodes, independent set, and matching. Regarding connectivity problems, the authors show that both variants are hard and that the approximation ratio of Con-Max is between 2 and $O(1 + \sqrt{k/\mathsf{Opt}})$, where k is the number of pebbles and Opt denotes the measure of an optimal solution. This result has been improved in [3], where the authors show that Con-Max can be approximated within a constant factor, more precisely 136. In [6] it is also shown that Con-Sum is not approximable within $O(n^{1-\epsilon})$ (for any positive ϵ), while it admits an approximation algorithm with ratio of $O(\min\{n \log n, k\})$ (where k is the number of pebbles). Moreover, they also provide an exact polynomial-time algorithm for Con-Max on trees.

More recently, in [7], a variant of the classical facility location problem has been studied. This variant, called *mobile facility location*, can be modelled as a motion planning problem and is approximable within $(3+\epsilon)$ (for any positive ϵ) if we seek to minimize the total movement [1]. On the other hand, a variant where the maximum movement has to be minimized admits a tight 2-approximation [1,6].

Finally, for Con and Clique, in [4] the authors present a set of improved (in)approximability results both for general and special classes of graphs, and moreover they also study the problem of moving pebbles to an independent set.

Our Problems and Results. More formally, our problems can be stated as follows. Let P be a simple polygon delimited by the set of vertices $V(P) = \langle v_1, \dots, v_n \rangle$, in this order. Let $A = \{p_1, \dots, p_k\}$ be a set of pebbles. Each pebble initially sits on a polygon vertex (multiple pebbles can occupy the same position). Thus, by $S = (s_1, \dots, s_k)$ we denote the initial configuration of the pebbles. Given a *target* vertex $v_i \in P$, we denote by $d(s_i, v_i)$ the length of a shortest path in P starting at s_i and ending at v_i. Such a shortest path is actually a vertex-to-vertex polygonal path, which is in compliance with our setting. Let $U = (u_1, \dots, u_k)$, with $u_i \in V$, denote the final configuration of the pebbles, and let $|d(S,U)| = \sum_{i=1}^{k} d(s_i, u_i)$, and $||d(S,U)|| = \max_{i=1,\dots,k}\{d(s_i, u_i)\}$. With a small abuse of notation, when in the final configuration all the pebbles sit on a same vertex u, we denote these quantities by $|d(S,u)|$ and $||d(S,u)||$. Finally, let $G(P)$ be the visibility graph of P. We study the following problems:

1. *Rendez-vous:* The questions we address are:
   (i) RV-Max: find $u^* = \arg\min_{u \in V(P)}\{||d(S,u)||\}$;
   (ii) RV-Sum: find $u^* = \arg\min_{u \in V(P)}\{|d(S,u)|\}$.
2. *Connectivity:* Let $\mathcal{C}$ denote the set of subsets of vertices of P which induce a connected subgraph in $G(P)$. Then, the questions we address are:
   (i) Con-Max: find $U^* = \arg\min_{U \in \mathcal{C}}\{||d(S,U)||\}$;
   (ii) Con-Sum: find $U^* = \arg\min_{U \in \mathcal{C}}\{|d(S,U)|\}$.
3. *Clique:* Let $\mathcal{K}$ denote the set of subsets of vertices of P which induce a clique in $G(P)$. Then, the questions we address are:

(i) Clique-Max: find $U^* = \arg\min_{U \in \mathcal{K}}\{||d(S,U)||\}$;
(ii) Clique-Sum: find $U^* = \arg\min_{U \in \mathcal{K}}\{|d(S,U)|\}$.

Besides the above problems, we also define the corresponding ones associated with the hop distance in P between v_i and v_j, say $h(v_i, v_j)$. An example of solutions for our problems w.r.t. both distance models is given in Fig. 1, while the results we present in the paper are summarized in Table 1, where by p and m we denote the size of the set of vertices of P initially occupied by pebbles and of the set of edges of $G(P)$, respectively.

Table 1. New (in bold) and old (with the reference therein) results for the various motion problems, where ρ denotes the best approximation ratio for the corresponding problem. All the inapproximability results hold under the assumption that $\mathsf{P} \neq \mathsf{NP}$.

	Max	Sum
RV	HD: **solvable in $O(pm)$ time**	HD: **solvable in $O(pm+k)$ time**
	ED: **solvable in $O(n \log n)$ time**	ED: **solvable in $O(pn+k)$ time**
Con	HD: NP-**hard; $\rho \geq 2$**; $\rho \leq 136$ [3]	HD: NP-**hard; $\rho \geq n^{1-\epsilon}$; $\rho \leq 1 + O(nk/\mathsf{Opt})$**
	ED: **polyAPX-hard**	ED: **polyAPX-hard**
Clique	HD: NP-**hard; $\rho \geq 3/2$**; $\rho \leq 1 + 1/\mathsf{Opt}$ [4]	HD: NP-**hard;** $\rho \leq 2$ [4]
	ED: open	ED: open

2 Rendez-vous

As far as the hop distance is concerned, RV-Max and RV-Sum have a naïve $O(pm)$ and $O(pm+k)$ time solution, respectively, whose improvement is a challenging open problem. Indeed, let $V(S)$ be the set of vertices of P initially occupied by the pebbles. Observe that a shortest hop-distance path is just a shortest path in the visibility graph $G(P)$ of P. Then, first of all we compute $G(P)$ in $O(n+m)$ time [9]. After, and only for RV-Sum, in $O(n+k)$ time we associate with each vertex v of P the multiplicity of pebbles initially sitting on it, say $\mu(v)$. Then, in $O(pm)$ time we find the p breadth-first search trees of $G(P)$ rooted at the vertices of $V(S)$. From these trees, it is easy to see that we are able in $O(pn)$ time to solve both problems, by computing for RV-Max and for RV-Sum respectively

$$x^* = \arg\min_{x \in V(P)}\{\max\{h(x,v)|v \in V(S)\}\},$$

$$x^* = \arg\min_{x \in V(P)}\Big\{\sum_{v \in V(S)} h(x,v) \cdot \mu(v)\Big\}.$$

Concerning the Euclidean distance, once again RV-Max and RV-Sum have a trivial $O(pn)$ and $O(pn+k)$ time solution, respectively, which work as follows. First, for RV-Sum only, in $O(n+k)$ time we associate with each vertex v of P the multiplicity $\mu(v)$. Then, for each vertex $v \in V(S)$ we can find its distance

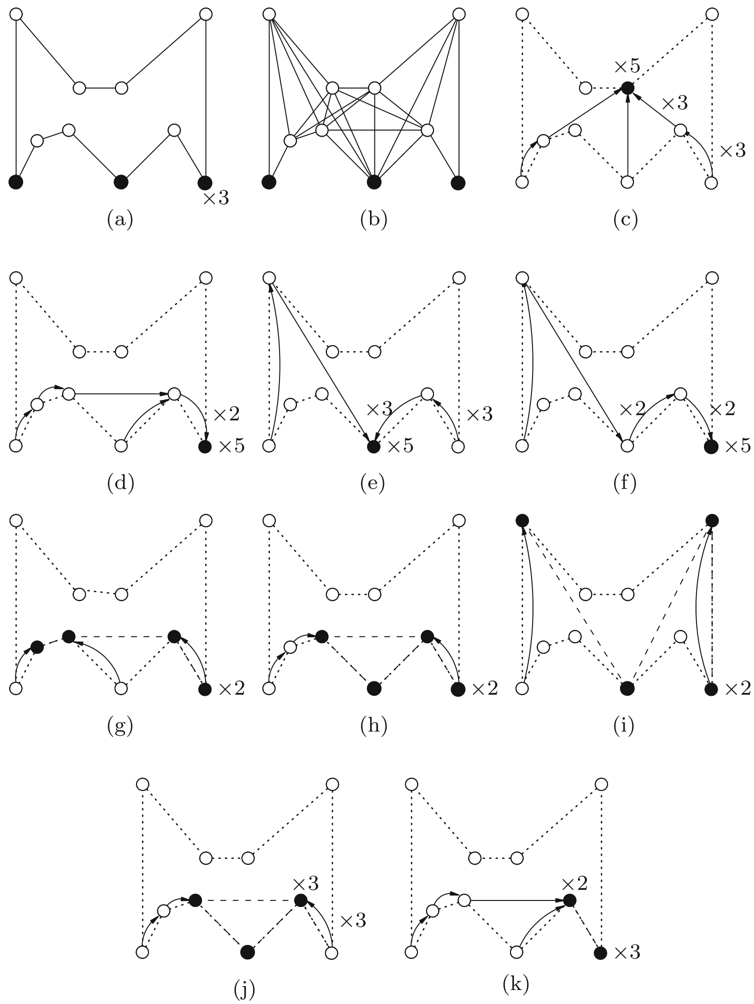

Fig. 1. An example of our studied problems for both HD and ED. Polygon P and its visibility graph $G(P)$ are shown in (a) and (b), respectively. Black vertices are occupied by pebbles, whose movement is depicted with directed paths. Optimal solutions for RV-Max and RV-Sum w.r.t. ED are shown in (c) and (d), respectively, while (e) and (f) show optimal solutions for the corresponding problems w.r.t. HD, respectively. Optimal solutions for Con-Max and Con-Sum w.r.t. ED are shown in (g) and (h), respectively, while an optimal solution for the corresponding problems w.r.t. HD is shown in (i). Finally, in (j) it is shown an optimal solution for Clique-Max w.r.t. to both ED and HD and Clique-Sum w.r.t. ED, while an optimal solution for Clique-Sum w.r.t. HD is shown in (k). Notice that dashed lines in (g–k) show the subgraph of $G(P)$ induced by the final position of the pebbles.

to all the other polygon vertices in $O(n)$ time [5,8]. Finally, similarly to the hop distance, in $O(pn)$ time we solve both problems. However, we now show that as far as the RV-Max problem is concerned, it is possible to provide an efficient $O(n \log n)$ time solution:

Theorem 1. *The* RV-Max *problem can be solved in* $O(n \log n)$ *time.*

Proof. Observe that in $O(n \log n)$ time (see [2]) we can compute the so-called *furthest-site geodesic Voronoi diagram* of $V(S)$ w.r.t. the Euclidean distance in P, i.e., a partition of P into a set of regions such that each region remains associated with the farthest point (in terms of Euclidean distance within P) in $V(S)$. Moreover, it can be shown [2] that the size of such a diagram is $O(n)$, and that given the diagram, for each vertex of P we can find in $O(1)$ time the farthest point in $V(S)$, i.e., the farthest pebble. Finally, we select the vertex for which the farthest pebble is closest, and we gather the pebbles there. □

3 Connectivity

3.1 Con-Max

Concerning Con-Max, let us start by focusing on the hop distance. Then, we are able to prove the following.

Theorem 2. *The* Con-Max *problem w.r.t. the hop distance is* NP*-hard.*

Proof. We show the NP-hardness by reduction from the NP-complete 3-Sat problem. In 3-Sat, we are given a set $X = \{x_1, \ldots, x_\eta\}$ of η variables, a set $Y = \{c_1, \ldots, c_m\}$ of m disjunctive clauses over X, each containing exactly three literals (i.e., a variable or its negation), and we want to find a truth assignment $\tau : X \to \{0, 1\}$ satisfying the conjunction of the clauses in Y. For a given instance $\mathcal{I}$ of 3-Sat, we build an instance $\mathcal{I}'$ for the Con-Max problem as follows: we build a simple polygon P, illustrated in Fig. 2, consisting of 2η *literal* vertices $V_L = \{x_1, \bar{x}_1, \ldots, x_\eta, \bar{x}_\eta\}$, η *assignment* vertices $V_A = \{a_1, \ldots, a_\eta\}$, $2\eta + 2m$ *gate* vertices $V_G = \{g_1, g'_1, \ldots, g_{\eta+m}, g'_{\eta+m}\}$, and $5m$ *clause* vertices $V_C = \{c_{11}, c_{12}, c_{13}, p_1, q_1, \ldots, c_{m1}, c_{m2}, c_{m3}, p_m, q_m\}$. Polygon P is so constructed such that, among the others, the following visibility constraints hold:

- literal vertices see each other reciprocally;
- each assignment vertex a_i can see only $g_i, g'_i, x_i, \bar{x}_i$;
- each clause vertex c_{ij} can see only $g_{\eta+i}, g'_{\eta+i}, p_i, q_i$, the other two clause vertices in its clause, and the literal vertex corresponding to the jth literal of its clause;
- each clause vertex p_i can see only $c_{i1}, c_{i2}, c_{i3}, q_i$.

Then, we put a pebble in each assignment vertex, and a pebble in each p_i, $i = 1, \ldots, m$, so the number of pebbles is $k = \eta + m$.

We now show that the 3-Sat instance $\mathcal{I}$ has a satisfying truth assignment iff there exists a solution for $\mathcal{I}'$ having maximum hop distance of 1. One direction

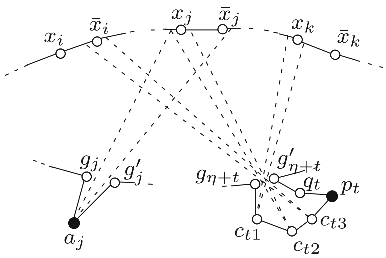

Fig. 2. The polygon P used for proving the NP-hardness of Con-Max problem w.r.t. the hop distance. Pebbles sit initially on black vertices.

is immediate. Given a satisfying assignment τ, we indeed define the following movement: each pebble in an assignment vertex moves to the appropriate literal verified by τ, while each pebble in a clause moves to any clause vertex seeing a verified literal. In this way, the pebbles originally sitting on assignment vertices will form a clique in the visibility graph $G(P)$, while the remaining pebbles are connected to (i.e., see) exactly an occupied literal vertex.

Concerning the other direction, suppose that there is a solution for $\mathcal{I}'$ having value at most 1. We show that such a solution can be transformed in polynomial time into a satisfying assignment for $\mathcal{I}$. First of all, notice that by construction of P, each assignment pebble must be moved to either an associated gate vertex or to an associated literal vertex to guarantee mutual visibility among assignment pebbles. Then, observe that in a single hop a pebble in a clause can see a literal vertex only if it moves to either of the three clause vertices that can see the respective literal vertices. Thus, to guarantee connectedness among assignment and clause pebbles, it is required that at least one of these three literals is occupied by an assignment pebble. Hence, the satisfying assignment for $\mathcal{I}$ is given exactly by the placement onto these literal vertices of the assignment pebbles that will guarantee the connectedness with the clause pebbles. Notice that some of the assignment pebbles may need not move to any literal vertex (i.e., the corresponding variable is not instrumental to guarantee the satisfiability of $\mathcal{I}$), and so they could simply move to an associated gate vertex in order to be connected to the assignment pebbles which moved towards the literal vertices. For these pebbles, we arbitrarily assign a value to the associated variables.

It remains to prove that P can be constructed in polynomial time. It suffices to show that P can be embedded on integer grid points with polynomial area and using a polynomial number of algebraic operations, similarly to the approach used in [10]. Let $r = \eta + m$. W.l.o.g., assume that $m = \Theta(\eta)$, and so $r = \Theta(\eta) = \Theta(m)$. Consider a circle C with radius $\Theta(r^2)$ centered at a point o. We position the literal vertices $x_1, \bar{x}_1, \ldots, x_\eta, \bar{x}_\eta$ on the upper side of C, regularly spaced. Let the angle (in radians) from o to any two contiguous vertices be $\Theta(1/r)$, and so the angle $\angle x_1 o \bar{x}_\eta$ can be less than a fraction of π (i.e., all the vertices stay on the upper side). Observe that in this way, the distance between two contiguous vertices is $\Theta(r)$. Now, position the assignment spikes on the lower-left side of C, so that the angle $\angle g_j a_j g'_j$ is $\Theta(1/r)$ and the distance between a_j and g_j, g'_j is $\Theta(r)$ (i.e., the distance between g_j and g'_j is $\Theta(1)$, and so we can actually put these vertices on the grid), and the visibility cone from a_j towards the upper side of C includes only the literal vertices $x_j, \bar{x}_j$ (indeed, the projection of the cone on the upper side of C has length $\Theta(r)$). Notice that each spike has area $\Theta(r)$. Moreover, again we can guarantee that the angle $\angle g_1 o g'_\eta$ is less than a fraction of π. Let us now consider the set of vertices in a clause, along with the associated gate vertices. Let $\{g_{\eta+t}, g'_{\eta+t}, c_{t1}, c_{t2}, c_{t3}, p_t, q_t\}$ be this set of vertices for the tth clause. We will embed these points on an $r \times r$ grid drawn at the lower-right side of C. Let o_t be the center of such a grid. We draw a circle C_t centered at o_t of radius $\Theta(r)$, and we append it to C by the gate vertices $g_{\eta+t}, g'_{\eta+t}$. We let the angle $\angle g_{\eta+t} o_t g'_{\eta+t}$ be $\Theta(1/r)$. Observe now that it is not hard to see that the angle $\angle g_{\eta+t} x g'_{\eta+t}$ is less than $\angle g_{\eta+t} o_t g'_{\eta+t}$, for any point x on the semi-circumference of C_t opposite to the gate vertices, since any such point is farther from $g_{\eta+t}, g'_{\eta+t}$ than o_t. So, the visibility cone from any such point towards the upper side of C has an angle $O(1/r)$ and then a projection on C of length $O(r)$. Thus, it includes a portion of C which is in the order of the distance between two contiguous literal vertices. Then, we place c_{t1}, c_{t2}, c_{t3} on the projection through the midpoint of $g_{\eta+t}, g'_{\eta+t}$ of the respective associated literal vertex. Finally, we suitably deform C_t so as to put p_t, q_t in such a way that p_t can only see c_{t1}, c_{t2}, c_{t3} and q_t. Again, the angle $\angle g_{\eta+1} o g'_{\eta+m}$ is less than a fraction of π. It can now be verified that this construction gives the desired polygon, and its area is $\Theta(r^4)$. □

Thus, since the problem is hard already when the optimal solution costs 1, we immediately have the following:

Corollary 1. *For any $\epsilon > 0$, the* Con-Max *problem w.r.t. the hop distance cannot be approximated within $2 - \epsilon$, unless* P = NP.

Moreover, the following implication is also easy to prove:

Corollary 2. *Deciding whether* Con-Max *admits a solution with at most h hops is* NP*-complete, for any $h \geq 1$.*

Proof. Case $h = 1$ follows directly from Theorem 2. For $h > 1$, it suffices to suitably modify the polygon P in Fig. 2 in such a way that the pebbles need to move for $h - 1$ steps in order to see the literal and the clause vertices. □

Concerning the approximability, we recall that in [3] the authors provide a 136-approximation for the very same problem on general unweighted graphs, which can therefore be applied to visibility graphs as well.

The above NP-hardness proof can be modified in order to show that the general Con-Max problem with Euclidean distances is NP-hard as well.

Theorem 3. *The* Con-Max *problem is* NP*-hard.*

Proof. We show the NP-hardness again by reduction from 3-Sat. For a given instance $\mathcal{I}$ of 3-Sat, we build an instance $\mathcal{I}'$ for Con-Max as follows: we build a simple polygon P with 2η *literal* vertices $V_L = \{x_1, \bar{x}_1, \ldots, x_\eta, \bar{x}_\eta\}$, η *assignment* vertices $V_A = \{a_1, \ldots, a_\eta\}$, $2m$ *gate* vertices $V_G = \{g_1, g'_1, \ldots, g_m, g'_m\}$, and $3m$ *clause* vertices $V_C = \{c_{11}, c_{12}, c_{13}, \ldots, c_{m1}, c_{m2}, c_{m3}\}$. Polygon P is so constructed such that, among the others, the following visibility constraints hold (see Fig. 3):

– literal and assignment vertices see each other reciprocally;
– each clause vertex c_{ij} can see only $g_{\eta+i}, g'_{\eta+i}$, the other two clause vertices in its clause, and the literal vertex corresponding to the jth literal of its clause.

Then, we put a pebble in each assignment vertex, and a pebble in each clause vertex, so the number of pebbles is $k = \eta + 3m$. Let us see how polygon P is actually constructed in polynomial time and with polynomial area. Let $r = \eta + m$. W.l.o.g., assume that $m = \Theta(\eta)$, and so $r = \Theta(\eta) = \Theta(m)$. Consider a circle C with radius $\Theta(r^3)$ centered at a point o. We position each triple of vertices $x_i, a_i, \bar{x}_i$ on the upper side of C, regularly spaced at a distance $\Theta(r)$. Then, we let the angle (in radians) from o to any two contiguous triplets be $\Theta(1/r)$ (i.e., the distance between two contiguous triplets is $\Theta(r^2)$). In this way, the angle $\angle x_1 o \bar{x}_\eta$ can be less than a fraction of π, and then assume that all these vertices lie in the $[\pi/4, 3\pi/4]$ sector. Let us now consider the set of vertices in a clause, along with the associated gate vertices. Let $\{g_t, g'_t, c_{t1}, c_{t2}, c_{t3}\}$ be this set of vertices for the tth clause. We will embed these points on an $r^2 \times r^2$ grid drawn at the lower side of C, in the $[5\pi/4, 7\pi/4]$ sector. Let o_t be the center of such a grid. We draw a circle C_t centered at o_t of radius $\Theta(r^2)$, and we append it to C by the gate vertices g_t, g'_t. We let the angle $\angle g_t o_t g'_t$ be $\Theta(1/r^2)$ (i.e., the distance between g_t and g'_t is $\Theta(1)$, and so we can actually put these vertices on the grid). Then, the visibility cone from any point on the semi-circumference of C_t opposite to the gate vertices towards the upper side of C has an angle of $O(1/r^2)$ and then a projection on C of length $O(r)$. Thus, it sees a portion of C including a single literal vertex. Then, we place c_{t1}, c_{t2}, c_{t3} on the projection through the midpoint of g_t, g'_t of the respective associated literal vertex. Notice that by construction, these vertices will lie in the lower side of C_t, and so they will be at $\Theta(r^2)$ distance from the respective gate vertices. It can now be verified that this construction gives the desired polygon, and its area is $\Theta(r^6)$.

Then, it is not hard to see that the 3-Sat instance $\mathcal{I}$ has a satisfying truth assignment iff there exists a solution for $\mathcal{I}'$ having maximum distance $\Theta(r)$. One direction is immediate. Given a satisfying assignment τ, we indeed define the following movement: each pebble in an assignment vertex moves to the appropriate literal verified by τ, while each pebble in a clause stands still. In this way, the pebbles originally sitting on assignment vertices will form a clique in the visibility graph $G(P)$, while for each clause there is at least a pebble connected to a literal vertex satisfying the clause, and so the other pebbles in the clause will remain connected to it. Notice that the maximum movement is $\Theta(r)$.

Concerning the other direction, suppose that there is a solution for $\mathcal{I}'$ having value $\Theta(r)$. We show that such a solution can be transformed in polynomial

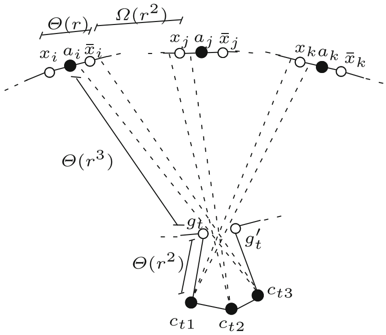

Fig. 3. The polygon P used for proving the NP-hardness of Con-Max, along with a specification of distances among vertices. Pebbles sit initially on black vertices.

time into a satisfying assignment for $\mathcal{I}$. First of all, notice that by construction of P, no assignment pebble can move beyond the adjacent literal vertices, and similarly no pebble in a clause can move to the associated gate vertices. Then, in order to guarantee connectedness, we have that each assignment vertex must move to a literal vertex, and moreover there must be at least a pebble in each clause seeing a literal vertex occupied by a pebble. Thus, this corresponds to a satisfying assignment for $\mathcal{I}$. $\square$

The above result has a very strong implication:

Corollary 3. Con-Max *is not approximable within any polynomial, unless* P $=$ NP.

Proof. Observe that the construction of the polygon P given in Theorem 3 can be inflated as follows: for any integer $k > 2$, we let the circle C have radius $\Theta(r^{k+1})$, we let the distance between two contiguous triplets be $\Theta(r^k)$, we embed each clause on an $r^k \times r^k$ grid, and we finally let the angle $\angle g_t o_t g'_t$ be $\Theta(1/r^k)$. It can now be verified that this construction gives a polygon of area $\Theta(r^{2(k+1)})$ for which an optimal solution of cost $\Theta(r)$ exists iff there is a satisfying assignment, while any approximate solution will require a pebble to be moved by $\Omega(r^k)$. Hence, since $r = \Theta(n)$, the claim follows. $\square$

3.2 Con-Sum

Concerning Con-Sum, the reduction shown in Theorem 2 can be modified to prove the following two results:

Theorem 4. *The* Con-Sum *problem w.r.t. the hop distance is* NP*-hard.*

Proof. We use the same construction as in Theorem 2. The claim is that the instance $\mathcal{I}$ of 3-SAT is satisfiable iff there is a solution for the instance $\mathcal{I}'$ of Con-Sum of cost at most $\eta + m$. Given a truth assignment, the existence of a solution of cost $\eta + m$ is immediate, since we have shown how to move every pebble at most by one to obtain connectedness. Now assume that we have a solution U with total movement of $h := ||d(S, U)|| \leq \eta + m$. First of all, we show that $h < \eta + m$ is unfeasible, and so it must be $h = \eta + m$.

For the sake of contradiction, assume that $h < \eta + m$ hops are enough. This means that there is at least a pebble that does not move. But then observe that the distance in the visibility graph $G(P)$ of P between any two initial positions of pebbles is at least 3, and so to be visually connected to other pebbles, a pebble that stands still asks for (at least) another pebble being moved by at least 2 hops. Moreover observe that each vertex of P guards at most a single vertex on which pebbles initially sit, and so no pebble which moves for at least 2 hops can be visually connected to more than one pebble which remained still. From this, we have that to guarantee connectedness, it must be $h \geq \eta + m$, a contradiction.

Then, let $h = \eta + m$. If each pebble has moved, we are done, since this implies that each clause pebble is connected in $G(P)$ to a literal pebble, and therefore we can compute (in polynomial time) a truth assignment for $\mathcal{I}$ by using the same arguments used in Theorem 2. Otherwise, assume this is not the case, and so there is at least one pebble that remained still. We will show that U can be modified into another solution U' such that (i) U' still has total movement h, and (ii) every pebble moves exactly one step in U'. The modification of the solution U is quite simple. Let H be the (connected) subgraph of the visibility graph induced by the final positions of the pebbles in U. Moreover, let p be a pebble sitting in the node v and which did not move in U. Consider a node v' which is adjacent to v in H onto which a pebble p' sits. In order to reach v', as explained before pebble p' moved by $t \geq 2$ hops. Moreover, observe that by construction the set of vertices which are visible from v is a subset of the set of vertices which are visible from v'. Then, we modify U as follows. We move p from v to v', and we move p' backwards by one step w.r.t. its path towards v'. In this way, the movement of p' is now $t - 1 \geq 1$, all the vertices which where guarded by p' are now guarded by p, and p and p' are connected. So the new pebble configuration is still connected and the total movement remains h. By proceeding in this way, we will arrive to the aimed configuration U'. □

Corollary 4. *For any* $0 < \epsilon < 1$*, the* Con-Sum *problem w.r.t. the hop distance cannot be approximated within* $n^{1-\epsilon}$*, unless* P=NP*.*

Proof. We adapt the reduction of Theorem 2 as follows: we modify the gadgets of the assignment vertices and of the clauses by adding $2N$ vertices and $N + 1$ pebbles for each gadget, where $N = (\eta + m)^{2/\epsilon - 1}$ (see Fig. 4).

Then, it can be shown that (see also the proof of Theorem 4):

(i) if there exists a satisfying truth assignment for $\mathcal{I}$, then there exists a solution for $\mathcal{I}'$ having total movement of $\eta + m$;
(ii) if there exists a solution for $\mathcal{I}'$ with total movement less than or equal to N, then there exists a satisfying truth assignment for $\mathcal{I}$. Indeed, as the total movement is less than or equal to N, a pebble placed on vertex a_j or vertex p_t has been moved by at most 1 (otherwise, all the other N pebbles placed in the same gadget would have been moved by at least 1).

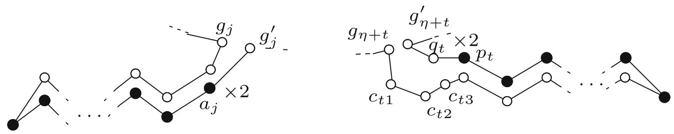

Fig. 4. The assignment and clause gadgets for proving the inapproximability of CON-SUM w.r.t. the hop distance. Pebbles sit initially on black vertices. Vertex a_j and vertex p_t contain two pebbles each.

Since 3-SAT is NP-complete and $n = \Theta\big((\eta + m)^{2/\epsilon}\big)$, the approximation ratio of any polynomial time algorithm for the CON-SUM problem must be at least

$$\frac{N}{\eta + m} = \frac{(\eta + m)^{2/\epsilon - 1}}{\eta + m} = (\eta + m)^{2/\epsilon - 2} = \Omega(n^{\epsilon/2})^{2/\epsilon - 2} = \Omega(n^{1-\epsilon}).$$

□

On the positive side, we have the following:

Theorem 5. *The* CON-SUM *problem w.r.t. the hop distance can be solved optimally up to an additive term of* $O(nk)$.

Proof. It suffices to observe that any solution which will bring all the pebbles to sit on a same vertex cannot require more than n additional hops for each of the k pebbles w.r.t. an optimal solution. □

On the other hand, when we consider the Euclidean distance, CON-SUM becomes much harder, as the following two results show:

Theorem 6. *The* CON-SUM *problem is* NP*-hard.*

Proof. The NP-hardness follows again by reduction from 3-SAT, by slightly modifying the construction given in Theorem 3. More precisely, we let the circle C have radius $\Theta(r^4)$, we let the distance between two contiguous triplets be $\Theta(r^3)$, we embed each clause on an $r^3 \times r^3$ grid, and we finally let the angle $\angle g_t o_t g'_t$ be $\Theta(1/r^3)$. It can now be verified that this construction gives a polygon of area $\Theta(r^8)$, for which it can be shown that there exists a satisfying assignment for 3-SAT iff there exists a solution for CON-SUM of costs $\Theta(r^2)$. □

Corollary 5. CON-SUM *is not approximable within any polynomial, unless* P=NP.

Proof. Observe that the construction of the polygon P given in Theorem 6 can be inflated as follows: for any integer $k > 3$, we let the circle C have radius $\Theta(r^{k+1})$, we let the distance between two contiguous triplets be $\Theta(r^k)$, we embed each clause on an $r^k \times r^k$ grid, and we finally let the angle $\angle g_t o_t g'_t$ be $\Theta(1/r^k)$. It can now be verified that this construction gives a polygon of area $\Theta(r^{2(k+1)})$ for which an optimal solution costs $\Theta(r^2)$ (and can be found in polynomial time iff P=NP), while any approximate solution will require a pebble to me moved by $\Omega(r^k)$. Hence, since $r = \Theta(n)$, the claim follows. □

4 Clique-Connectivity

As far as the clique-connectivity problems are concerned, we are able to provide results only for the hop distance case, while the Euclidean case remains open.

4.1 Clique-Max

Concerning Clique-Max, it is easy to see that the problem can be solved optimally up to an additive term of 1, by just guessing a vertex belonging to an optimal solution onto which all the pebbles are moved (see [4]). In spite of that, the problem is hard, as proven in the following:

Theorem 7. *The* Clique-Max *problem w.r.t. the hop distance is* NP*-hard.*

Proof. We suitably modify the reduction of Theorem 2. So, the reduction is still from the 3-Sat problem. For a given instance $\mathcal{I}$ of 3-Sat, we build an instance $\mathcal{I}'$ for the Clique-Max problem as follows: we build a simple polygon P consisting of 3η *literal* vertices $V_L = \{x_1, \check{x}_1, \bar{x}_1, \ldots, x_\eta, \check{x}_\eta, \bar{x}_\eta\}$, $5m$ *clause* vertices $V_C = \{c_{11}, c_{12}, c_{13}, p_1, q_1, \ldots, c_{m1}, c_{m2}, c_{m3}, p_m, q_m\}$, $2m$ *gate* vertices $V_G = \{g_1, g'_1, \ldots, g_m, g'_m\}$, 3 *obstacle* vertices y_1, y_2, y_3, and finally five *auxiliary* vertices $z_1, z_2, z_3, z_4, \bar{p}$. Polygon P is so constructed that, among the others, the following visibility constraints hold (see Fig. 5):

– every literal vertex x_i sees all the other literal vertices but $\bar{x}_i$, and vice versa;
– each clause vertex c_{ij} can see only $g_{\eta+i}, g'_{\eta+i}, p_i, q_i$, the other two clause vertices in its clause, and the literal vertex corresponding to the jth literal of its clause;
– each clause vertex p_i can see only $c_{i1}, c_{i2}, c_{i3}, q_i$;
– gate vertices cannot see auxiliary vertices due to the obstacle made up by y_1, y_2, y_3;
– $\bar{p}$ can see only z_3 and z_4, z_3 and z_4 see each other and can see only $z_1, z_2, \bar{p}$, while z_1 and z_2 can see all the literal vertices but not the gate vertices.

Then, we put a pebble in each clause vertex p_i, $i = 1, \ldots, m$, and a pebble in $\bar{p}$.

We now show that the 3-Sat instance $\mathcal{I}$ has a positive answer iff there exists a solution for $\mathcal{I}'$ having maximum hop distance of 2. One direction is simple. Given a satisfying assignment τ, we indeed define the following movement: each pebble in a clause moves first (with a single hop) to any clause vertex seeing a verified literal, and then it reaches the corresponding literal vertex with an additional hop. Moreover, we move the pebble in $\bar{p}$ to z_1. In this way, the assignment vertex will form a clique in the visibility graph $G(P)$, and each pebble makes 2 hops.

Concerning the other direction, suppose that there is a solution for $\mathcal{I}'$ having value at most 2. We show that such a solution can be transformed in polynomial time into a satisfying assignment for $\mathcal{I}$. First of all, notice that the pebble in $\bar{p}$ must be either in z_1 or z_2. Moreover, by construction, since the final positions of the pebbles induce a clique, it must be the case that every pebble is on a literal

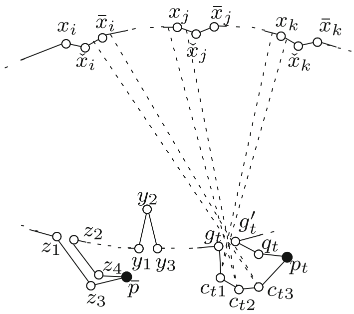

Fig. 5. The polygon P used for proving the NP-hardness of CLIQUE-MAX w.r.t. the hop distance. Pebbles sit initially on black vertices.

vertex. Indeed, the only vertices that can be reached by 2 hops from p_t and which are visible by z_1 or z_2 are the literal vertices associated with the tth clause. Moreover, we cannot have two pebbles on x_i and $\bar{x}_i$, because these two vertices cannot see each other. Hence, the final positions of the clause pebbles define a truth assignment for the formula. Notice that it can be the case that there is no pebble in x_j nor in $\bar{x}_j$ (i.e., the corresponding variable is not instrumental to guarantee the satisfiability of $\mathcal{I}$). In this case we assign an arbitrary value to x_j.

It remains to show that P can be constructed in polynomial time. Actually, the construction is similar to that used in Theorem 2, and so we leave it to the reader. We just point out that the angle $\angle \bar{x}_i x_i \check{x}_i$ must be $\Theta(1/r^2)$, in order to hide only $\bar{x}_i$ to x_i (i.e., the distance between $\check{x}_i$ and the ray passing through $x_i, \bar{x}_i$ will be $\Theta(1)$). □

Since the problem is hard already when the optimal solution costs 2, we have:

Corollary 6. *For any $\epsilon > 0$,* CLIQUE-MAX *w.r.t. the hop distance cannot be approximated within $3/2 - \epsilon$, unless* P = NP.

Moreover, the following implication is also easy to prove:

Corollary 7. *Deciding whether* CLIQUE-MAX *admits a solution with at most h hops is* NP*-complete, for any $h \geq 2$.*

Proof. Case $h = 2$ follows directly from Theorem 7. For $h > 2$, it suffices to suitably modify the polygon P used in Theorem 7 in such a way that the pebbles need to move for $h - 1$ steps in order to see the literal and the clause vertices. □

4.2 Clique-Sum

Concerning CLIQUE-SUM, once again we restrict ourselves to the hop distance case. First of all, notice that in this case the problem is 2-approximable [4]. However, it turns out that a slight modification of the reduction used for CLIQUE-MAX yields the following:

Theorem 8. *The* CLIQUE-SUM *problem w.r.t. the hop distance is* NP*-hard.*

Proof. We use the same construction as in Theorem 7. W.l.o.g. we assume that in the instance of 3-SAT the mth clause c_m contains only the variable x_η (either negate or not), and that x_η occurs only in c_m. We claim that $\mathcal{I}$ is satisfiable iff $\mathcal{I}'$ admits a solution of total movement at most $2(m+1)$. One direction is immediate, since we have proved that if $\mathcal{I}$ has a satisfying truth assignment then we can move the pebbles towards a clique with maximum movement of 2. Now, assume that we have a solution of total movement of at most $2m+2$. We will show that every pebble must move by at least 2 hops to guarantee the clique constraint (and so actually at least $2m+2$ hops are needed). This immediately implies the claim, since this means that each pebble moves exactly 2 steps, and so we can compute (in polynomial time) a truth assignment for $\mathcal{I}$ by using the same arguments used in Theorem 7. First of all, observe that the hop distance between any two p_t and $p_{t'}$ is at least 4 (i.e., it is 4 if c_t and $c_{t'}$ share a literal, otherwise is 5). Moreover, the hop distance between any p_t and $\bar{p}$ is 5. Finally, for our assumption about instance $\mathcal{I}$, the hop distance between p_t and p_m is 5, for every $t \neq m$. Let h be the movement of a pebble p. In order to move all the pebbles in a feasible configuration, we have that two pebbles have been moved by at least $4-h$ hops, and the remaining $m-2$ pebbles have been moved by at least $3-h$ hops. Summing up over all the pebbles, the total movement is at least $3m+2-(m-1)h$, which is less than or equal to $2m+2$ only when $h \geq \lceil m/(m-1) \rceil$, i.e., $h \geq 2$. □

5 Discussion and Open Problems

Motion planning in a constrained environment is susceptible of a deep investigation in several respects. Here we have limited our attention to planar vertex-to-vertex motion in a simple polygon and with the objective of achieving very basic configurations, but it is easy to imagine more challenging scenarios. For instance, notice that relaxing the assumption that pebbles have to start, turn, and stop at vertices only will make the planning task substantially more difficult. On the other hand, a simplifying yet very interesting setting is that in which the constraining polygon is orthogonal.

As far as the problems in our setting are concerned, we point out that it remains open to understand the computational properties of CLIQUE (both MAX and SUM) w.r.t. the Euclidean distance. Moreover, establishing whether CLIQUE-MAX for the hop distance is hard already when an optimal solution costs 1 is very intriguing: indeed, such a case retains a strong connection with the CLIQUE DOMINATING SET (CDS) problem (i.e., deciding whether a graph has a dominating clique). For general graphs, it is known that this problem is NP-complete, while it is unknown whether CDS is NP-complete for visibility graphs. Notice that if CDS was NP-complete for visibility graphs, we would have the NP-hardness of CLIQUE-MAX already restricted to instances where an optimal solution costs 1 (indeed, it suffices to consider instances with a pebble in each vertex). Conversely, if we prove that Clique-Max is polynomially solvable for $h = 1$, then this implies that CDS for visibility graphs is also decidable in polynomial time. Finally, we feel that an improvement of the 136-approximation algorithm for Con-Max w.r.t. the hop distance might be possible, by exploiting the special nature of visibility graphs.

Acknowledgements. The authors wish to thank an anonymous referee for her/his insightful comments, which helped us in improving the paper.

References

1. Ahmadian, S., Friggstad, Z., Swamy, C.: Local-search based approximation algorithms for mobile facility location problems. arXiv preprint:1301.4478 (2013)
2. Aronov, B., Fortune, S., Wilfong, G.T.: The furthest-site geodesic Voronoi diagram. Discrete Comput. Geom. **9**, 217–255 (1993)
3. Berman, P., Demaine, E.D., Zadimoghaddam, M.: $O(1)$-approximations for maximum movement problems. In: Goldberg, L.A., Jansen, K., Ravi, R., Rolim, J. (eds.) RANDOM 2011 and APPROX 2011. LNCS, vol. 6845, pp. 62–74. Springer, Heidelberg (2011)
4. Bilò, D., Gualà, L., Leucci, S., Proietti, G.: Exact and approximate algorithms for movement problems on (special classes of) graphs. In: Moscibroda, T., Rescigno, A.A. (eds.) SIROCCO 2013. LNCS, vol. 8179, pp. 322–333. Springer, Heidelberg (2013)
5. Chazelle, B.: Triangulating a simple polygon in linear time. Discrete Comput. Geom. **6**, 485–524 (1991)
6. Demaine, E.D., Hajiaghayi, M., Mahini, H., Sayedi-Roshkhar, A.S., Oveisgharan, S., Zadimoghaddam, M.: Minimizing movement. In: Proceedings of the 18th ACM-SIAM Symposium on Discrete Algorithms (SODA'07), pp. 258–267 (2007)
7. Friggstad, Z., Salavatipour, M.R.: Minimizing movement in mobile facility location problems. ACM Trans. Algorithms 7(3), article **28**, 1–22 (2011)
8. Guibas, L.J., Hershberger, J., Leven, D., Sharir, M., Tarjan, R.E.: Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons. Algorithmica **2**, 209–233 (1987)
9. Hershberger, J.: An optimal visibility graph algorithm for triangulated simple polygons. Algorithmica **4**(1), 141–155 (1989)
10. Lin, Y.-W., Skiena, S.S.: Complexity aspects of visibility graphs. Int. J. Comput. Geom. Appl. **5**(3), 289–312 (1995)
11. Prencipe, G., Santoro, N.: Distributed algorithms for autonomous mobile robots. In: Navarro, G., Bertossi, L., Kohayakawa, Y. (eds.) TCS 2006. IFIP, vol. 206, pp. 47–62. Springer, Heidelberg (2006)
12. Suri, S., Vicari, E., Widmayer, P.: Simple robots with minimal sensing: From local visibility to global geometry. Int. J. Robot. Res. **27**(9), 1055–1067 (2008)