A primer on covering spaces for robot exploration

Ari Blumenthal
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801 USA
ablumen2@illinois.edu

May 16, 2011

Abstract

The notion of *covering spaces*, with its roots in algebraic topology, can be used to study many aspects of robot exploration and environmental modeling. We will look at the problem of a mobile robot exploring an environment with an omnidirectional detection sensor and see how covering spaces can be useful in constructing a map of the environment. Then we will begin to quantify the limitations of the sensor by exploring the *badness* of a given environment.

1 Introduction

The problem of exploring and mapping environments with robots is an important problem in the robotics community. With technology today growing smaller, faster and cheaper than ever before, a major focus has been tackling this challenge with simple inexpensive devices. As such it is as important as ever to solve these problems with minimalistic sensor models and provide efficient means for modeling complex environments.

One such minimalistic approach looks at a robot with a single pebble. A natural problem that arises is determining whether or not such robot can learn the environment. An algorithm for this is provided in [1] and shows that it is indeed possible in strongly connected graphs. A further problem explores whether or not an environment has any holes. Results in [3] have found that a single robot with a pebble is able to make these distinctions.

In this paper, we will continue this minimalistic approach and look at how with simple sensor models, it is possible for robots to describe complex environments full of landmarks with simpler, more efficient models. To do so, we will draw on previous work done in algebraic topology [2], in which the notion of covering spaces has provided a useful tool. More specifically, we will first define our use of covering spaces and the problem more generally in Section 2. Then we will go on to look at some examples of these simpler spaces in Section 3. Later we will introduce a metric for comparing different spaces (Section 4).

2 Problem definition

Let an **environment**, written E, be a subset of $\mathbb{R}^2$. Let $G \subset E$ be a set of **landmarks**, where each $g \in G$ is a point in E. We will consider a robot as a point free to move about the environment. As such, we describe its state space as $X \subset \mathbb{R}^2$. The robot will be equipped with a single omnidirectional sensor capable of detecting and reporting the set of labels visible to the robot at its current location.

Let L be the set of all valid class labels. With this sensor model, our robot can recognize the labels of visible landmarks, but not how many of an individual label are present. Therefore, the sensor mapping will be a function $h : X \to \mathcal{P}(L)$. An important concept for us to think about then is the preimage of our sensor mapping, $h^{-1}(s) = \{x \in X \mid s = h(x)\}$. In other words, given a set of labels $s \in \mathcal{P}(L)$, $h^{-1}(s)$ is the set of all possible points in the environment where our robot could currently be located while having its sensor output be s.

We say two points p and q are **visibly connected** if there exists a path from p to q such that for every point p' in that path $h(p) = h(p') = h(q)$. A visibly connected region m is **maximal** if d is another visibly connected region and $d \supseteq m \Rightarrow d = m$. Two maximal visibly connected regions, d_1 and d_2 are **adjacent** if $p \in d_1$ and $q \in d_2$ implies that there exists a path from p to q and a single point r on that path, such that for every point p' before r, $h(p') = h(p)$ and every point q' after r, $h(q') = h(q)$.

When working with these regions, it often makes intuitive sense to instead think about the dual graph. That is, we create the **visibility graph**, written Ψ, where the set of all vertices in Ψ, written $V(\Psi)$, contains all maximal visibly connected regions. There is an edge between $d_1, d_2 \in V(\Psi)$, written $d_1 d_2 \in E(\Psi)$ for each border d_1 shares with d_2. It is important to note two key features of these regions: (1) they are always distinct ($\forall d_1, d_2 \in V(\Psi), d_1 \cap d_2 = \emptyset$) and (2) the entire set of regions covers E ($\bigcup_{d \in V(\Psi)} d = E$). Also, one key feature of this graph is that it is inherently planar, since the environment of which it is based is $\mathbb{R}^2$.

Let Ψ and $\tilde{\Psi}$ be the visibility graphs for X and $\tilde{X}$, respectively. A **covering space** of a state space X is a space $\tilde{X}$ together with a map $p : \tilde{X} \to X$ that satisfies the condition that there exists a set of maximal visibly connected regions $V(\Psi) = \{x_\alpha\}$ of X such that for each α, $p^{-1}(x_\alpha)$ is a disjoint union of maximal visibly connected regions in $\tilde{X}$. In addition, we say that a covering space is **proper** if the following conditions hold:

1. Let $x_\alpha \in V(\Psi)$ and let $Y = \{x_\beta : x_\alpha x_\beta \in E(\Psi)\}$. If $p(a) = x_\alpha$ then for each $y \in Y$ there exists a unique $b \in V(\tilde{\Psi})$ such that $p(b) = y$ and $ab \in E(\tilde{\Psi})$.

2. For every sequence $A = \{a_1, a_2, \ldots\}$ of maximal visibly connected regions where each $a_i \in V(\tilde{\Psi})$, if $a_i a_{i+1} \in E(\tilde{\Psi})$ and $a_i a_{i+1} \neq a_{i-1} a_i$ for all $a_i \in A$, then there exists another sequence $D = \{x_1, x_2, \ldots\}$ where each $x_i \in V(\Psi)$ and the following hold true for all $a_i \in A$: (a) $p(a_i) = x_i$, (b) $x_i x_{i+1} \in E(\Psi)$, and (c) $x_i x_{i+1} \neq x_{i-1} x_i$.

3. Given sensor mappings $h : X \to \mathcal{P}(L)$ and $\tilde{h} : \tilde{X} \to \mathcal{P}(L)$ for our robot, then for all $a \in \tilde{X}$ and $x \in X$, $p(a) = x$ if and only if $\tilde{h}(a) = h(x)$

Note that a proper covering space requires that (1) that there is a strong local restraint which requires every intersection in the covering space to have the same number of paths entering and exiting it as in the original space, (2) that each path a robot could take in the covering space $\tilde{X}$ by 'just walking forward' can be taken in the original space X by doing the same, and (3) sensor output is conserved by the mapping p. In some situations it might make more sense to weaken one of these restraints, in which case proper covering spaces could be defined differently.

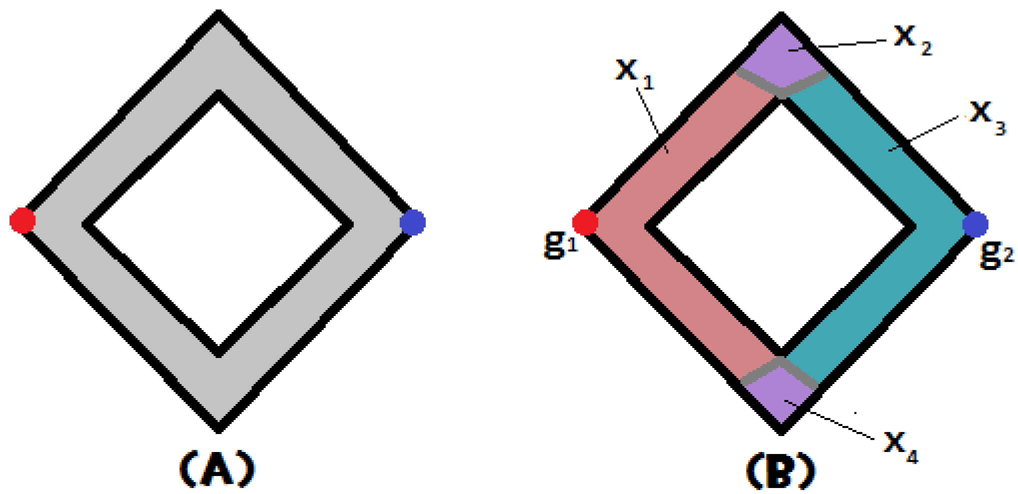

Figure 1: (a) Hallway with a closed loop (b) Closed loop again with maximal visibly connected regions shown

3 Introduction to different environments

In order to fully understand these spaces, it is important to explore some example spaces and associated covering spaces.

3.1 Closed loop

The first environment we will look at is that of a single closed loop and can be seen in Figure 1(a). The set $G = \{g_1, g_2\}$ contains two landmarks, each with a unique label (shown in the figure by their color). Here the set of possible labels $L = \{$“blue”, “red”$\}$. The valid locations for the robot to move about the environment are shown by the areas shaded gray. Let Ψ_0 be the visibility graph for this loop. In Figure 1(b), one can see that there are four maximal visibly connected regions, so $V(\Psi_0) = \{x_1, x_2, x_3, x_4\}$. When the robot is in x_1, only the red landmark is visible, so $h(x_1) = \{$“red”$\}$. Similarly, when the robot is in x_3, only the blue landmark is visible, so $h(x_3) = \{$“blue”$\}$. In both x_2 and x_4 either landmark is visible, so $h(x_2) = h(x_4) = L$. Also, for each $x_i \in V(\Psi_0), x_i x_{(i+1)\%4} \in E(\Psi_0)$ and $x_i x_{(i+3)\%4} \in E(\Psi_0)$.

Now, let us consider which environments could be covering spaces for

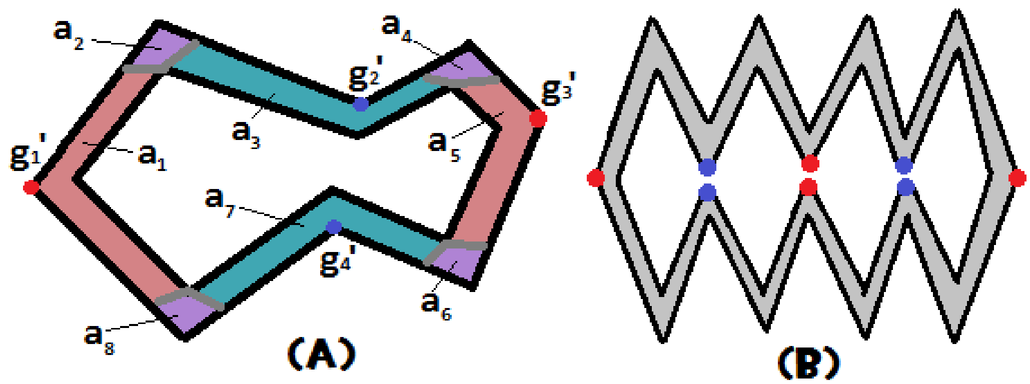

Figure 2: (a) A longer closed loop. (b) An even longer closed loop

our simple closed loop. The first ones we will look at are larger loops with more than two landmarks. Consider the loop seen in Figure 2(a). The set $G' = \{g'_1, g'_2, g'_3, g'_4\}$ contains four landmarks, each with one of the two labels from L. Let Ψ_1 be the visibility graph for this loop. The robot can move between any of the shaded maximal visibly connected regions. The set of all such regions $V(\Psi_1) = \{a_1, a_2, \ldots, a_8\}$. For all $a_i \in V(\Psi_1)$, there are two edges adjacent to it: $a_i a_{(i+7)\%8}$ and $a_i a_{(i+1)\%8}$. If we let $\tilde{h} : \tilde{X} \to \mathcal{P}(L)$ be the sensor mapping for our robot moving around this environment, then $\tilde{h}(a_1) = \tilde{h}(a_5) = \{$"red"$\}$, $\tilde{h}(a_3) = \tilde{h}(a_7) = \{$"blue"$\}$, and $\tilde{h}(a_k) = L$ for k even.

Let X be the state space of a robot travelling around the original closed loop and let $\tilde{X}$ be the state space of a robot traveling around the loop in Figure 2(a). Let $p : \tilde{X} \to X$ be defined as $p(a_i) = x_{i\%4}$. Therefore, each x_i has two values mapping to it, a_i and a_{2i}. So, for every maximal visibly connected region $x_i \in X$, $p^{-1}(x_i) = a_i \cup a_{2i}$ and $a_i \cap a_{2i} = \emptyset$. Hence, $\tilde{X}$ is a covering space for X.

But we can do better still! $\tilde{X}$ is a *proper* covering space of X. To confirm this, let us check the three necessary conditions:

1. Let $x_i \in V(\Psi_0)$. Then $Y = \{x_{(i+1)\%4}, x_{(i+3)\%4}\}$. Since $p^{-1}(x_i) = \{a_i, a_{2i}\}$, we have two regions to check. First let us check a_i. Since $a_i a_{i+1} \in E(\Psi_1)$ and $p(a_{i+1}) = x_{(i+1)\%4}$, pick a_{i+1} for $x_{(i+1)\%4}$. For $x_{(i+3)\%4}$, we pick $a_{(i+7)\%8}$, because $a_i a_{(i+7)\%8} \in E(\Psi_1)$ and $p(a_{(i+7)\%8}) = x_{((i+7)\%8)\%4} = x_{(i+3)\%4}$. Similarly, for a_{2i}, we pick x_{2i-1} for $x_{(i+3)\%4}$ and

$x_{(2i+1)\%8}$ for $x_{(i+1)\%4}$.

2. Let A be an arbitrary sequence that satisfies the necessary conditions. That is, the robot is either moving clockwise or counter-clockwise around the larger closed loop. If it is moving clockwise, then $A = \{a_i, a_{(i+1)\%8}, a_{(i+2)\%8}, \ldots\}$. Consider then

   $$D = \{p(a_i), p(a_{(i+1)\%8}), p(a_{(i+2)\%8}), \ldots\} = \{x_{i\%4}, x_{(i+1)\%4}, x_{(i+2)\%4}, \ldots\}$$

   Now we have to check three three conditions. Well, for (a), if x_k and a_k are the kth elements of D and A, respectively, then $p(a_k) = x_k$ for all $a_k \in A$ and $x_k \in D$ by construction. For (b), if $x_k = x_{m\%4}$ for some $m \in 0, 1, 2, 3$, then $x_{k+1} = x_{(m+1)\%4}$ and clearly $x_{m\%4}x_{(m+1)\%4} \in E(\Psi_0)$. For (c), since the robot is always moving clockwise around the loop, it is impossible for $x_k x_{k+1} = x_{k-1}x_k$. Therefore the three conditions are satisfied for all clockwise sequences. Now suppose the robot is moving counter-clockwise around the larger closed loop. Then A is of the form $\{a_i, a_{(i+7)\%8}, a_{(i+6)\%8}, \ldots\}$, and we pick $D = \{x_{i\%4}, x_{(i+3)\%4}, x_{(i+2)\%4}, \ldots\}$. Similarly we could show that our set D satisfies the three conditions.

3. This is obviously true, since $\tilde{h}(a_1) = \tilde{h}(a_5) = h(x_1) = \{\text{“red”}\}$, $\tilde{h}(a_3) = \tilde{h}(a_7) = h(x_1) = \{\text{“blue”}\}$, and both labels are visible everywhere else.

As can be seen in Figure 2(b), we can go further and make the loop even larger. In this case the previous methods for mapping maximal visibly connected regions to that of the original closed loop are still valid. In fact we could make the loop arbitrarily big, by increasing the number of ‘notches’ and landmarks in the loop, so that if the robot moved around the larger loop, it remains consistent with what it would see if it moved around the smaller loops. This allows us to create an infinite set of loops which are all covering spaces for our original closed loop!

But a question remains of whether constructing environments in this way provides the only method of generating covering spaces for the original closed loop. In fact, that is not the case. Consider now an infinite path like that seen in Figure 3. For this environment, the set G contains an infinite number of landmarks, still with only the two labels from L. If we let Ψ_2 be the visibility graph for the infinite path, then the set of all maximal visibly connected regions $V(\Psi_2) = \{\ldots, a_{-2}, a_{-1}, a_0, a_1, a_2, \ldots\}$ is infinite and for each $a_k \in V(\Psi_2)$, $a_{k-1}a_k \in E(\Psi_2)$ and $a_k a_{k+1} \in E(\Psi_2)$.

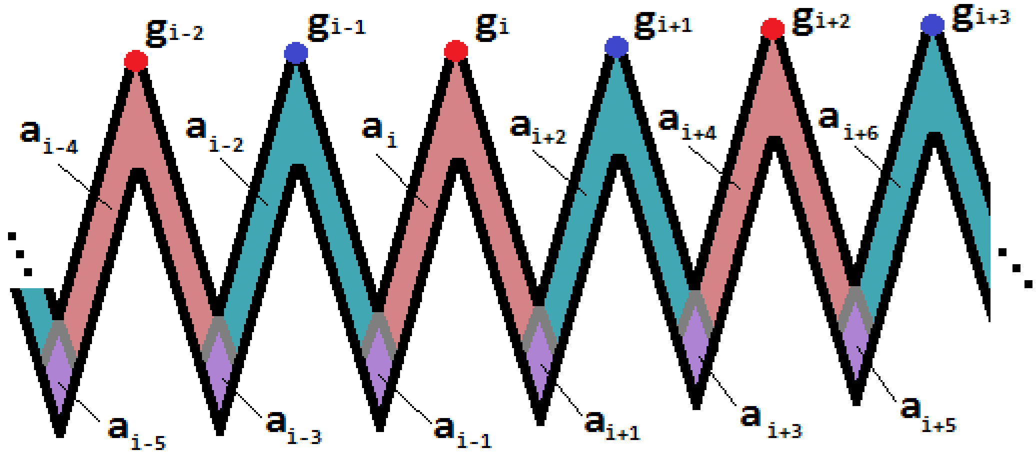

Figure 3: An infinite path. Also a covering space for the closed loop and a really long stroll for the robot.

Let X again be the state space of a robot travelling around the original closed loop, and let X_2 be the state space for a robot traveling in the infinite path. Let $p : X_2 \to X$ be defined as $p(a_i) = x_{i\%4}$. Now, each x_i has an infinite number of regions mapping to it. That is, for every maximal visibly connected region $x_i \in X, p^{-1}(x_i) = \bigcup_{n \in \mathbb{Z}} a_{4n+i}$ and for any $m, n \in \mathbb{Z}$ such that $m \neq n$, $a_{4m+i} \cap a_{4n+i} = \emptyset$. Hence, X_2 is a covering space for X. We could also show that X_2 is even a proper covering space of X following the same methods used for the larger closed loop.

So, now that we've seen a couple examples of proper covering spaces, it makes sense to look at an example of a space that initially seems like it might be a proper covering space, but fails. Consider the path seen in Figure 4(a), which extends to infinity in one direction, but terminates in the other. This is quite similar to the previous infinte path that we saw; however, the fact that this one terminates in one direction prevents it from being a proper covering space for X. Lets look at why.

Let Φ_3 be the visibility graph for this new path. Then $V(\Phi_3) = \{a_1, a_2, a_3, \ldots\}$ and for all $i \geq 2$, $a_i a_{i-1} \in E(\Phi_3)$. Let X_3 be the state space for a robot traveling in this path. Let $p : X_3 \to X$ be defined as $p(a_i) = x_{i\%4}$. So we have

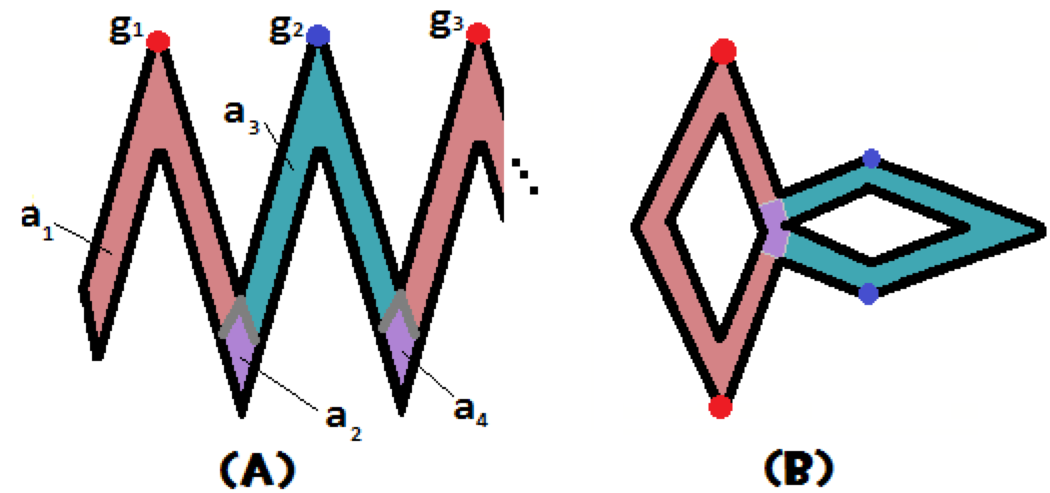

Figure 4: (a) An infinite path that terminates in one direction (b) The double loop.

for all $x_i \in X, p^{-1}(x_i) = \bigcup_{n \in \mathbb{N}} a_{4n+i}$ and for any $m, n \in \mathbb{N}$ such that $m \neq n$, $a_{4m+i} \cap a_{4n+i} = \emptyset$. So, X_3 is a covering space of X. However, it is *not* a proper covering space. It fails when we look at the local restraint. Consider $x_1 \in V(\Phi)$. Then let Y $= \{x_2, x_4\}$. Since $p(a_1) = x_1$ then for each element of Y there must exist a unique $b \in V(\Phi_3)$ that is adjacent to a_1 and maps to it. However, there is only one region adjacent to a_1, namely a_2, so we cannot find a second region to map to the other element of Y. Hence, X_3 is not a proper covering space of X.

Now that we have a better understanding of covering spaces, lets look at how one might quantitatively compare different spaces.

4 Measure of an environment

Let $\mathcal{O} = \{s \in \mathcal{P}(L) \mid h^{-1}(s) \neq \emptyset\}$. We call this the **visibility set** of a pair (G, E), since it is the set of sets of labels that a robot could potentially see if it were to travel to every point in E. Recall that for a visibility graph Φ, that $V(\Phi)$ is the set of all maximal visibly connected regions. Let the **badness** of a pair (G, E) be $\mathcal{B}(G, E) = \dfrac{|V(\Phi)|}{|\mathcal{O}|}$. So, when $\mathcal{B}(G, E) > 1$, we know that there are multiple maximal visibly connected regions that share the same set of labels $s \in \mathcal{P}(L)$. Ideally, if we needed to store or keep track of an environment in a robot's memory, we would want to minimize the amount of data repetition. Therefore, by comparing badness we provide a useful means of distinguishing the usefulness of different spaces.

Lets look at a couple examples. Consider the double loop seen in Figure 4(b). If Φ is the visibility graph for this environment, then $|V(\Phi)| = 3$, since there are three disjoint maximal visibly connected regions. $\mathcal{O} = \{\{\text{“red”}\}, \{\text{“blue”}\}, \{\text{“red”}, \text{“blue”}\}\}$, so $|\mathcal{O}| = 3$. Thus, $\mathcal{B}(G, E) = 3/3 = 1$.

In Figures 5 and 6, we provide several covering spaces for the double loop (the ten proofs have been omitted for the sake of brevity). In each case, $|\mathcal{O}| = 3$; however, the number of maximal visibly connected regions varies greatly from one to the next. For example, in covering space (1), $|V(\Phi_1)| = 6$, so $\mathcal{B}(G, E) = 6/3 = 2$. For (3), $|V(\Phi_3)| = 9$, so $\mathcal{B}(G, E) = 9/3 = 3$. For (5), $|V(\Phi_5)| = 12$, so $\mathcal{B}(G, E) = 12/3 = 4$. On the other hand, for spaces (7)-(10), the number of maximal visibly connected regions is infinite, so $\mathcal{B}(G, E) = +\infty$.

In fact, for all $n \geq 1$, it is possible to construct a covering space of the double loop with a badness of n. Covering spaces (2) and (4) provide an interesting intuition into this construction. For (2), the outer boundary of the environment is a quadrilateral and the resulting badness is 2. For (4), it is a hexagon with a badness of 3. Suppose we want to construct an environment that produces a covering space with an arbitrary badness of k. We need only start with a path in the shape of a $2k$-gon with a blue landmark on every other vertex. Then, we connect each vertex without a blue landmark to the adjacent vertices without blue landmarks and add a red landmark in each added path. This will result in an environment with k intersections from which blue and red landmarks are visible, k regions from which just a blue landmark is visible, and k regions from which only red is visible. This gives a total of $3k$ maximal visibly connected regions and a badness of k.

5 Conclusions

As we've seen, covering spaces can be a powerful tool for finding simpler spaces that seem identical to larger, more complicated environments. As

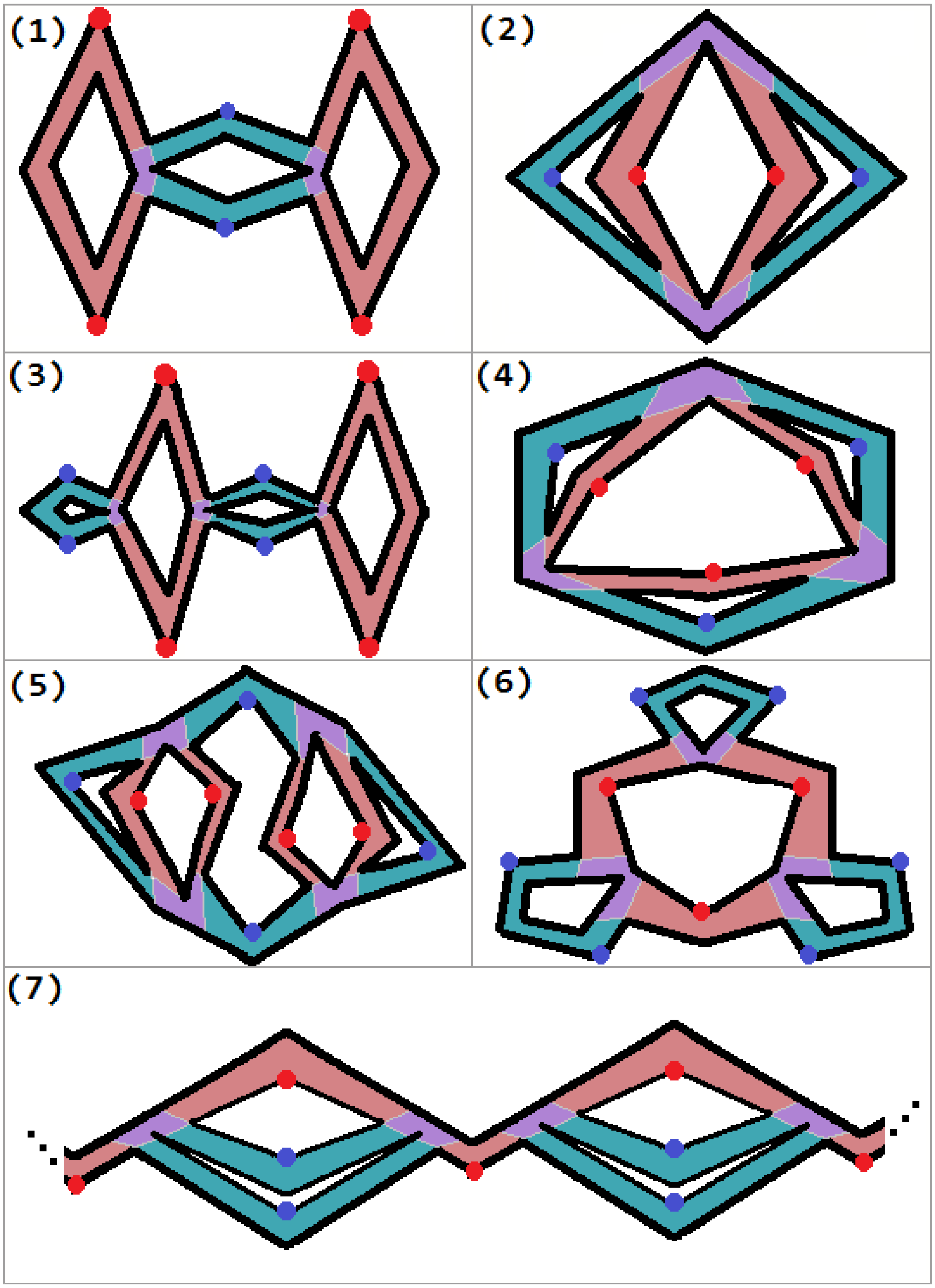

Figure 5: Several Covering space for the double loop.

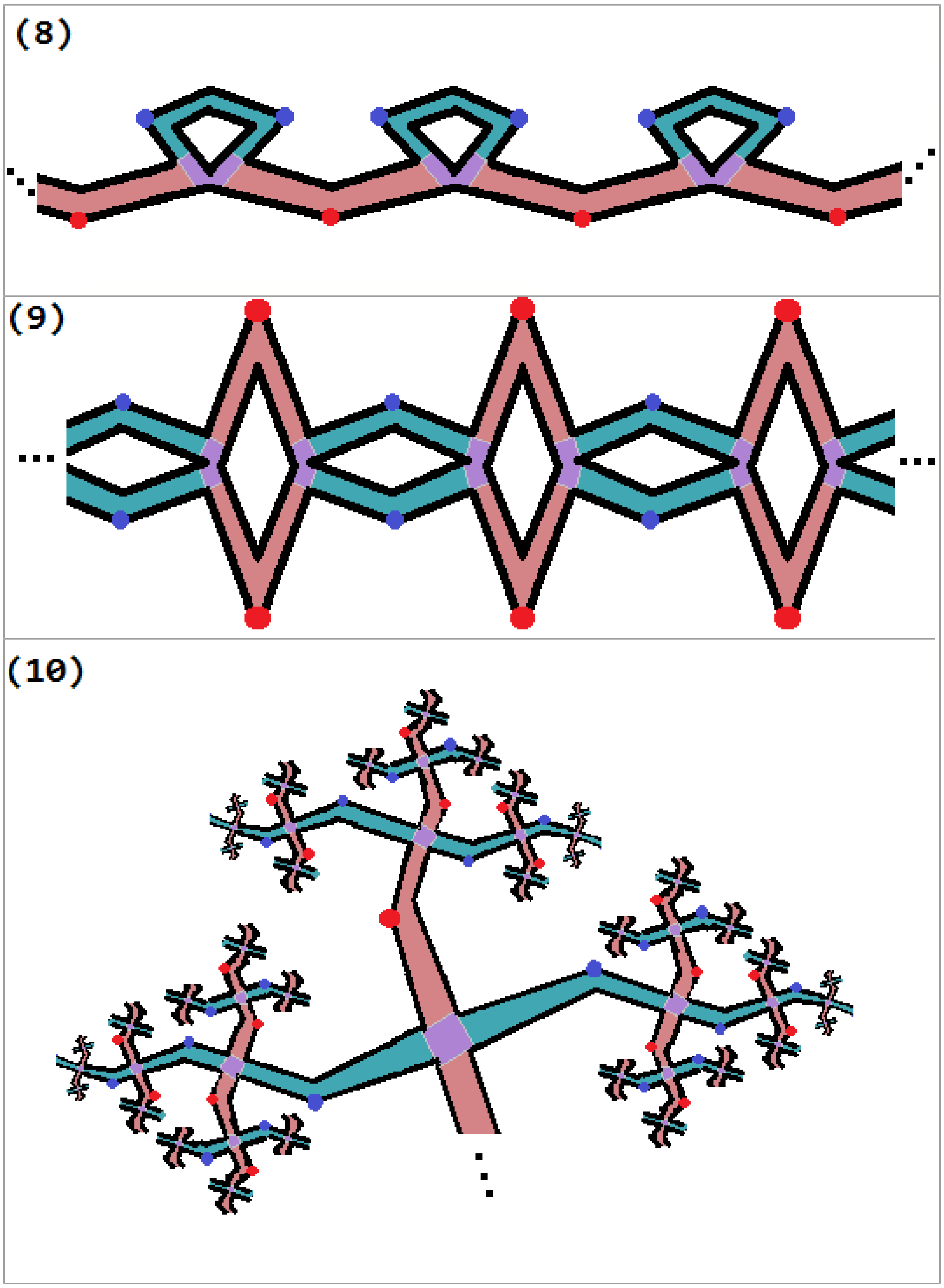

Figure 6: Several Covering space for the double loop (cont.)

such, this tool could prove useful in many problems associated with robotics and motion planning, such as counting targets or finding goals. One could even remove the dependence on landmarks and explore the spaces generated when multiple robots are released in an environment.

Finally, it would be quite useful to study further the design of complex environments, which are covering spaces for much simpler spaces. This would allow robots to move around and work in the complex environment, while more easily performing computations on a simpler map of the environment. In this way, it may be possible to decrease the difficulty of normally challenging tasks.

References

[1] M. Bender, A. Fernandez, D. Ron, A. Sahai, and S. Vadhan. The power of a pebble: Exploring and mapping directed graphs. *Information and Computation*, 176:1–21, 2002.

[2] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[3] S. Suri, E. Vicari, and P. Widmayer. Simple robots with minimal sensing: From local visibility to global geometry. *International Journal of Robotics Research*, 2008.