Synthesizing Algorithms to Avoid an Obstacle with a Swarm of Robots (Additional content)

Due to the lack of space, we moved in this extension of our paper every result that did not fit in the main paper. We first tackle in Section I the proof obligations allowing us to ensure that the generated algorithms perform the perpetual exploration on every (big enough) grid and for every initial configuration. Then we prove some impossibility results in Section II. And finally, we detail the rules of the five main generated extensions of $\mathcal{A}_b^1$ in Section III.

I. Proof of Perpetual Exploration

The two first steps of our methodology generated a set $\mathbb{A}_1$ of 5 valid extensions of $\mathcal{A}_b^1$ and a set $\mathbb{A}_2$ 115 valid extensions of $\mathcal{A}_b^2$. However, due to the nature of the property, we cannot automatically prove that all those algorithms satisfy the perpetual exploration for every grid. We now propose a generic scheme of proof for any algorithm $\mathcal{A} \in \mathbb{A}_1 \cup \mathbb{A}_2$ that extends the results obtained by simulation. In the following, we thus prove Lemma 3 and Theorem 2 of the paper, respectively corresponding to the Proof Obligations 1 and 2 of our methodology. We first start by an intermediate result.

Let $i \in \{1, 2\}$ and $\mathcal{A} \in \mathbb{A}_i$. First, notice that since the rules of $\mathcal{A}_b^i$ are not modified and $\mathcal{A} \in \mathbb{A}_i$ is valid by construction, the properties of $\mathcal{A}_i$ on grids without obstacles are preserved, as mentionned by Lemma 4.

Lemma 4. *Let $i \in \{1, 2\}$. Let $\mathcal{A} \in \mathbb{A}_i$. For any grid $\mathcal{G}$ of size $\mathcal{C} \times \mathcal{L}$ where $\mathcal{C}, \mathcal{L} \geq 2$ without obstacle, the execution of $\mathcal{A}$ starting from any configuration of $\mathcal{I}_i$ satisfies the perpetual exploration of $\mathcal{G}$.*

Now, the simulations validated that any $\mathcal{A} \in \mathbb{A}_i$ performs perpetual exploration in every grid in $\mathbf{G}_i^{\mathtt{sim}}$ starting from any configuration in $\mathcal{I}_i^{\mathtt{sim}}$, see Lemmas 1 and 2. We extend this result by extending first the set of initial configurations to $\mathcal{I}_i$ (see Lemma 3), second the sizes of the grids (see Theorem 2).

Lemma 3. *Let $i \in \{1, 2\}$. Let $\mathcal{A} \in \mathbb{A}_i$, $\mathcal{G} \in \mathbf{G}_i^{\mathtt{sim}}$, $\gamma \in \mathcal{I}_e^i$. The execution of $\mathcal{A}$ starting from γ satisfies the perpetual exploration of $\mathcal{G}$.*

Proof. Let e be the execution of $\mathcal{A}$ starting from γ. If $\gamma \in \mathcal{I}_i^{\mathtt{sim}}$, e satisfies the perpetual exploration of $\mathcal{G}$, see Lemmas 1 and 2 (Section IV.C of the main paper). Otherwise, the robots eventually reach a configuration of $\mathcal{I}_i^{\mathtt{sim}}$ from which the previous case applies: they perform the perpetual exploration. Indeed, in γ the robots do not see the obstacle so they cannot distinguish $\mathcal{G}$ from a similar grid $\mathcal{G}'$ of same size but without any obstacle. So, until the obstacle enters the visibility range of one of the robot, the robots will behave in e just as they would do executing in $\mathcal{G}'$. By Lemma 4, the robots eventually arrive in one of the reachable configurations of $\mathcal{A}_i$ but with an obstacle in the visibility range of one robot. In other words, they are in a configuration of $\mathcal{I}_i^{\mathtt{sim}}$. $\square$

Theorem 2. *Let $size_1 = 7$ and $size_2 = 13$. Let $i \in \{1, 2\}$. Let $\mathcal{A} \in \mathbb{A}_i$. For every grid $\mathcal{G}$ of size $\mathcal{C} \times \mathcal{L}$, where $\mathcal{C}, \mathcal{L} \geq size_i$, with an obstacle in node (x, y) such that $ObsInTheMiddle(x, y, \mathcal{C}, \mathcal{L})$, for every configuration $\gamma \in \mathcal{I}_e^i$, the execution of $\mathcal{A}$ starting from γ satisfies the perpetual exploration of $\mathcal{G}$.*

Proof. We prove this result by induction on the grid size, namely on the pair $(\mathcal{C}, \mathcal{L})$. The induction proof is straightforward, therefore we only detail the base cases and induction steps.

Base cases: In the proof, the induction steps require to one dimension by two. That is why we consider all the grids of the set $\mathbf{G}_i^{\mathtt{sim}}$. Therefore, the property for the base cases is covered by Lemma 3.

In the rest of the proof, we denote by v_i the visibility range of robots in $\mathcal{A}_i$.

Induction step #1: Let $\mathcal{C} \geq size_i + 1$ and $\mathcal{L} \geq size_i$. If $\mathcal{A}$ perpetually explores every grid $\mathcal{G}$ of size $\mathcal{C} \times \mathcal{L}$ with an obstacle in (x, y) such that $ObsInTheMiddle(x, y, \mathcal{C}, \mathcal{L})$ then it is also the case in every grid $\mathcal{G}'$ of size $(\mathcal{C} + 2) \times \mathcal{L}$ with an obstacle in (x', y') such that $ObsInTheMiddle(x', y', \mathcal{C} + 2, \mathcal{L})$.

Proof of Induction step #1: It is possible to build $\mathcal{G}'$ from $\mathcal{G}$ by adding two additional columns in between the $(\mathcal{C} - (v_i + 1))^{\text{th}}$ and $(\mathcal{C} - (v_i + 2))^{\text{th}}$ columns if $2(v_i + 1) \leq x' \leq \mathcal{C} - 2(v_i + 1) - 1$ or the $(v_i + 1)^{\text{th}}$ and $(v_i + 2)^{\text{th}}$ columns otherwise. Notice that the robots perform a repetitive pattern while exploring the grid. In particular, when they are at distance at least $(v_i + 1)$ from the wall they are moving towards, the opposite wall (*e.g.* the left wall if they are moving towards the right wall), and the obstacle, the local configuration is the same before and after the execution of a round. Thus, the additional columns does not change the behavior of the robots: the robots exit the additional columns the same way they entered it.
Similarly, when they reach a wall and are at distance at least $(v_i + 1)$ from the side walls (*e.g.* the top and bottom wall if they reached the right wall), robots are in the same local configuration before and after exploring two columns (or rows, depending on their direction).

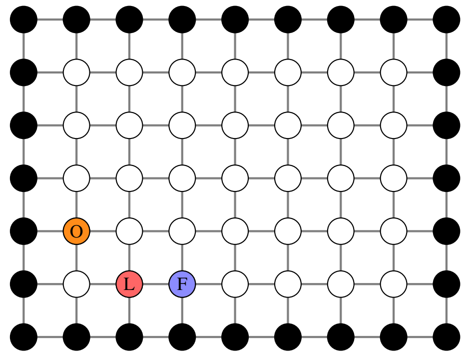


Fig. 1: Example of grid containing a dead-end obstacle.

Since they perform the perpetual exploration of $\mathcal{G}$, they will do so in $\mathcal{G}'$ too.

Induction step #2: Let $\mathcal{C} \geq size_i$ and $\mathcal{L} \geq size_i + 1$. If $\mathcal{A}$ perpetually explores every grid $\mathcal{G}$ of size $\mathcal{C} \times \mathcal{L}$ with an obstacle in (x, y) such that $ObsInTheMiddle(x, y, \mathcal{C}, \mathcal{L})$, then it is also the case in every grid $\mathcal{G}'$ of size $\mathcal{C} \times (\mathcal{L}+2)$ with an obstacle in (x', y') such that $ObsInTheMiddle(x', y', \mathcal{C}, \mathcal{L}+2)$.
Proof of Induction step #2: The proof is similar to the one of Induction step #1. Since we can build $\mathcal{G}'$ from $\mathcal{G}$ by adding two rows and $\mathcal{A}$ does the perpetual exploration of $\mathcal{G}$, it also perpetually explores $\mathcal{G}$'.

□

II. Impossibility Results

In this section, we prove some impossibility results that, due to the lack of space, could not be given in the main paper.

A. Dead-end Obstacles

We say that a grid contains a *dead-end obstacle* if the obstacle is next to a first wall and at distance two from a second wall. An example of such a grid is given on Fig. 1.

Lemma 5. *No extension algorithm of $\mathcal{A}_b^1$ with the same parameters (number of robots, colors, visibility range,* etc.*) can solve the perpetual exploration problem in a grid containing a dead-end obstacle.*

Proof. By contradiction, assume that some extension $\mathcal{A}$ of $\mathcal{A}_b^1$ solves the perpetual exploration in a grid $\mathcal{G}$ containing a dead-end obstacle. Without loss of generality, let assume that the dead-end obstacle is next to the left wall, at distance two from the bottom wall.

The initial configuration of $\mathcal{A}_b^1$ being locally-defined, there is an initial configuration such that the leader robot (of color L, without loss of generality) is next to the bottom wall at distance two from the left wall and the follower robot (of color F) is next to it, at distance three from the left wall. An example of such configuration is given in Fig. 1. In this situation, the robots do not see the obstacle and execute rules of $\mathcal{A}_b^1$ and move to the left. L is now in the dead-end and cannot execute rules of $\mathcal{A}_b^1$. There are two possibilities: L either remains idle or moves to the right to exit the dead-end. But F still does not see the obstacle, so it execute the rule of $\mathcal{A}_b^1$ and moves towards L. In both cases, there is a collision, leading to a contradiction. □

B. Impossibility with One Robot

The impossibility result of solving perpetual exploration of a grid with one robot is revisited in grids containing an obstacle. Proofs are heavily inspired from the ones without obstacles in [1]. We first recall straightforward result from the reachable configurations of a perpetual exploration algorithm.

Lemma 6. *Let $\mathcal{A} = (Col, \mathcal{I}, Rules)$ be an algorithm that solves perpetual exploration. For any (empty) location (i, j) in a grid, there is a configuration γ where (i, j) is occupied by a robot and the algorithm $\mathcal{A}' = (Col, \mathcal{I}', Rules)$ where $\mathcal{I}' = \mathcal{I} \cup \{\gamma\}$ also solves the perpetual exploration.*

Theorem 3. *The perpetual exploration problem is not solvable in grids containing an obstacle with only one robot, for any finite visibility range.*

Proof. Assume by contradiction, that $\mathcal{A}$ solves this problem with one robot. Let $\Phi > 0$ be the visibility range of the robot. Consider a grid $\mathcal{G}$ of size greater than $(2\Phi+5+\delta)\times(2\Phi+5+\delta)$, where $\delta \geq 0$ is the distance between the obstacle and the closest wall. By Lemma 6, we can assume, without loss of generality, that $\mathcal{A}$ solves the problem starting from the initial configuration where the unique robot is at distance at least $\Phi + 2$ from the obstacle and the walls. Thus, the robot does not see them. If the algorithm dictates the robot to stay idle, then the robot will stay idle forever, a contradiction. So, the robot moves toward one direction in the globally oriented view, say $d \in \{Up, Down, Left, Right\}$. Now, after the move, the robot is in the same situation: it sees neither a wall nor the obstacle. So, it will again move and the chosen destination in its local view will be the same. However, since the robot is self-inconsistent and the four possible destinations are indistinguishable, it may go in the opposite direction to d. Hence, there is a possible execution where the robot starts at a center of the grid and forever alternates between two positions. The perpetual exploration is not achieved in this execution, a contradiction. □

C. Impossibility with Two Robots and Less than Three Colors

The impossibility results of solving perpetual exploration of a grid with two robots with less than three colors are revisited in grids containing an obstacle. Again, proofs are heavily inspired from [1].

Lemma 7. *Let $\mathcal{A}$ be an algorithm that solves perpetual exploration problem with two robots and $\mathcal{G}$ a grid of size at least $(4\Phi + 9 + \delta) \times (4\Phi + 9 + \delta)$, where δ is the minimum distance between the obstacle and a wall and Φ is the visibility range.*

If a robot is at distance at least $2\Phi + 4$ from the obstacle and the walls then:

1. *the other robot is at distance at least $\Phi + 2$ from the obstacle and walls and*

2) *the two robots see each other.*

Proof. We proceed by contradiction. By Lemma 6 and without loss of generality, let consider a configuration γ_0 where a robot r is located at distance at least $2\Phi + 4$ from the obstacle and the walls. Let r' be the second robot.

Notice first that if r and r' see each other, then r' is at distance at least $\Phi + 2$ from the obstacle and walls. Assume now that the two robots never see each other.

Using similar arguments than for Theorem 3, there is an execution where r alternates between two locations ℓ_1 and ℓ_2 and never sees r'. Considers such an execution.

Let ℓ'_1 be a location at distance $\Phi+1$ from ℓ_1 and distance $\Phi + 2$ from ℓ_2. By construction of $\mathcal{G}$, ℓ'_1 is at distance at least $\Phi + 2$ from the walls and the obstacle. Moreover, ℓ'_1 is eventually visited since $\mathcal{A}$ solves the perpetual exploration problem. By construction, ℓ'_1 is necessarily visited first by r'. When the first visit of ℓ'_1 by r' occurs, neither robot see each other, nor a wall or the obstacle. Moreover, ℓ'_1 has a neighboring location, say ℓ'_2 at distance a least $\Phi+1$ from the walls and the obstacle. From that point, there is an execution where r' alternates between locations ℓ'_1 and ℓ'_2 since it has no landmarks to orient itself.

Hence, there exists a possible suffix of execution where all nodes but ℓ_1, ℓ_2, ℓ'_1 and ℓ'_2 are never visited again, a contradiction. □

Theorem 4. *The perpetual exploration problem is not solvable in grids containing an obstacle with only two robots and one color, for any finite visibility range.*

Proof. Assume by contradiction that $\mathcal{A}$ solves this problem with two robots and one color. Let $\Phi > 0$ be the visibility range.

Consider a grid of size at least $(4\Phi+9+\delta)\times(4\Phi+9+\delta)$, where δ is the minimum distance between the obstacle and a wall. By Lemmas 6 and 7, we can assume, without the loss of generality, that $\mathcal{A}$ solves perpetual exploration problem starting from an initial configuration γ_0 where a robot r is located at distance at least $2\Phi + 4$ from the obstacle and the walls and the other robot r' is within its visibility range.

Neither robot sees a wall or the obstacle in this initial configuration. Hence, their local views are the same and their destinations in the globally oriented view are opposite (they cannot stay idle or they would do so forever) and the distance between r and r' increases at each round. After at most $\lceil \frac{\Phi}{2} \rceil$ rounds, the robots are distant enough that they do not see each other. But even the closest robot from a wall or the obstacle cannot see it. From that point, using the same argument as in Theorem 3, we can construct a possible suffix of execution where each robot either stays idle or alternates between two nodes. Now, in that suffix, there are many nodes that are never visited, a contradiction. □

Theorem 5. *The perpetual exploration problem is not solvable in grids containing an obstacle with only two robots and two colors and visibility range one.*

Proof. Assume by the contradiction that an algorithm $\mathcal{A}$ this problem under this setting. Consider a grid $\mathcal{G}$ of size at least $14 + \delta \times 14 + \delta$, where δ is the minimum distance between the obstacle and a wall.

By Lemmas 6 and 7, we can assume, without the loss of generality, that $\mathcal{A}$ solves the perpetual exploration problem starting from an initial configuration γ_0 where a robot r is located at distance at least 6 from the obstacle and the walls and the other robot r' is located on a neighboring node. Moreover, they have different colors, since otherwise we have a contradiction using the same argument as in the proof of Theorem 4. Indeed, the argument remains valid even if robots can change their color because robots with the same views modify their color in the same way.

Let say that the color of r is B and the color of r' is R. Their local views are respectively V_1 and V_2 on Fig. 2. $\mathcal{A}$ outputs a single destination d_1 and d_2 for the local views V_1 and V_2, respectively.

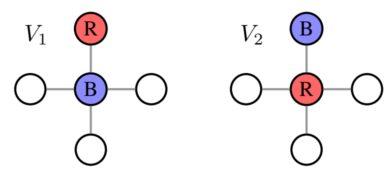


Fig. 2: Two possible views of two robots with colors R and B.

After one round, robots are either *(1)* isolated (*i.e.*, they do not see each other), *(2)* they have the same color, or *(3)* their views are V_1 and V_2 respectively (*n.b.*, the two robots still do not see the border of $\mathcal{G}$ nor the obstacle in this latter case). Let study these three cases:

(1). If robots are isolated, we can construct a possible execution where robots remain isolated forever and, depending on the algorithm, they either stay idle or alternate between two positions forever. Hence, in that case, perpetual exploration is not achieved, a contradiction.

(2). Now, if the robots have the same color after one round, then depending on the algorithm, we can construct an execution where they either stay idle or swap their position forever while maintaining their colors identical, or they become isolated after the second round. In the latter case, they still do not see the border of $\mathcal{G}$ nor the obstacle, so we can make the perpetual exploration fails, as previously explained.

(3). The only remaining case is the one where their views of the two robots are V_1 and V_2 after the first round. In this case, the movements of robots are periodic while they do not reach a wall or the obstacle. So, either robots alternate between two configurations, or the two robots move in a straight line until at least one robot reaches a wall or the obstacle. In the former case, we immediately obtain a contradiction. So, we focus now on the latter case. There exists executions where the robots move away from the obstacle and reach first a wall. Let consider such an execution.

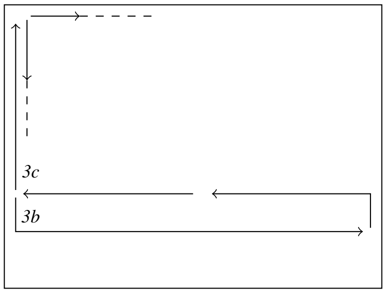


Fig. 3: The movement leading to a contradiction.

When a robot reaches a wall, there exists a constant S (independent of the size of the grid) such that the two robots can either *(3a)* remain indefinitely in a subgrid of size S, or *(3b)* remain in a subgrid of size S for a finite number of rounds, then leave the wall and move straight toward the opposite wall, or *(3c)* remain at distance at most S from the wall until they reach a corner. Fig. 3 illustrates the last two cases.
In Case *3a*, some nodes will not be visited if $\mathcal{G}$ is large enough, a contradiction.

In Case *3b*, if $\mathcal{G}$ is large enough, they will not encounter the obstacle while moving forward. Moreover, when the robots leave the wall they must be in the same relative positions as initially, rotated by angle π (to move in the opposite direction). Indeed, we have just seen that this is the only way the two robots can travel without seeing walls. Hence, when reaching the opposite wall, since they cannot distinguish between the two walls, the robots will perform the same turn, remains in a subgrid of size S for the same finite number of rounds, then leave the wall to move straight towards the first wall. All the movements are the same as the movement performed when reaching the first wall, but rotated by π. Hence, they will end up in the exact same initial position. The two robots will continue this periodic movement infinitely, leaving nodes unvisited if $\mathcal{G}$ is large enough, a contradiction.

Consider now Case *3c*. Again if $\mathcal{G}$ is large enough and the obstacle is far enough away from the wall, the robots will not encounter it. After following the wall (staying at distance at most S from it until reaching another wall), the robots reach the corner and again we have three possibilities:

- If they remain in a subgrid of size at most S' where S' is a constant (independent of the size of the grid), some nodes are never visited if $\mathcal{G}$ is large enough, a contradiction.
- If they follow a wall (either the same one in the opposite direction or the new encountered one) for a finite number of rounds, then leave the wall to move in straight line until reaching the opposite wall, there exists a constant S'' (independent of the size of the grid) such that the robots always stay at distance at most S'' from a wall, and when they reach a new wall, the same thing occurs again, so if $\mathcal{G}$ is large enough, some nodes that are far from all walls are never visited, leading to a contradiction.
- If they follow a wall until reaching another corner, exactly as in the previous case, some nodes that are far from the walls are never visited.

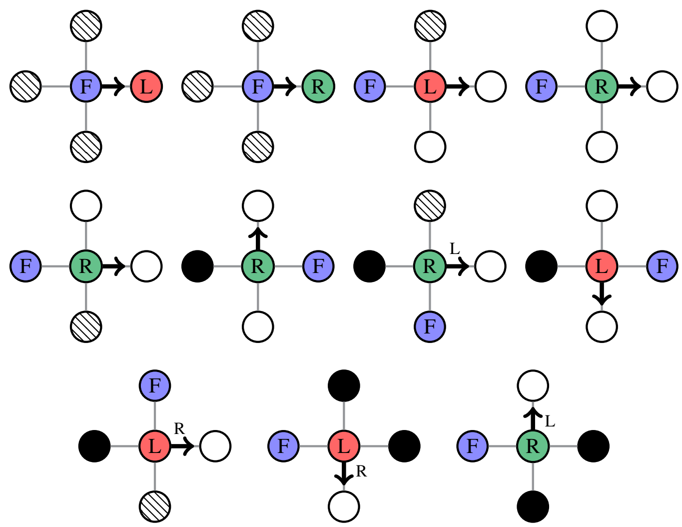
□

III. Generated Algorithms

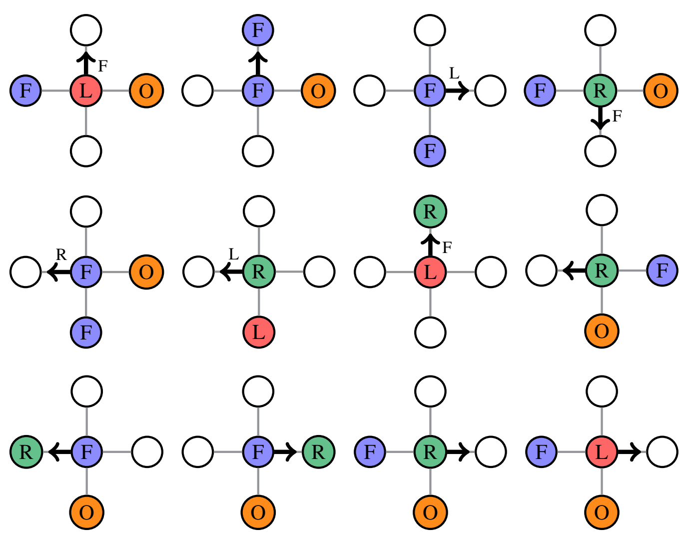
In this section, we present the rules of the five generated algorithms for $\mathcal{A}_b^1$, starting with the rules of $\mathcal{A}_b^1$ that are common to the five extensions.

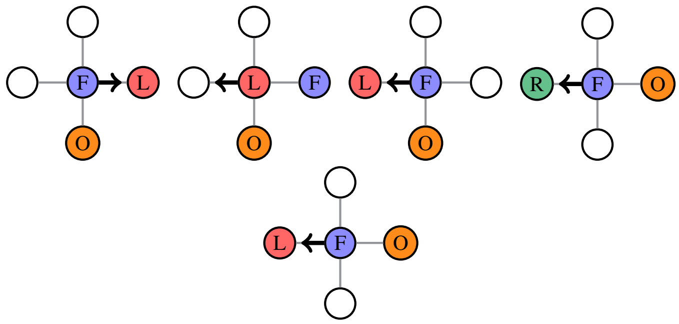
A. *Rules of $\mathcal{A}_b^1$*

The rules of $\mathcal{A}_b^1$ are presented in a compact way. On those rules, a hatched node represents an empty node or a wall, meaning that a robot will execute this rule if there is nothing or a wall in that place, but not if there is a robot or an obstacle.

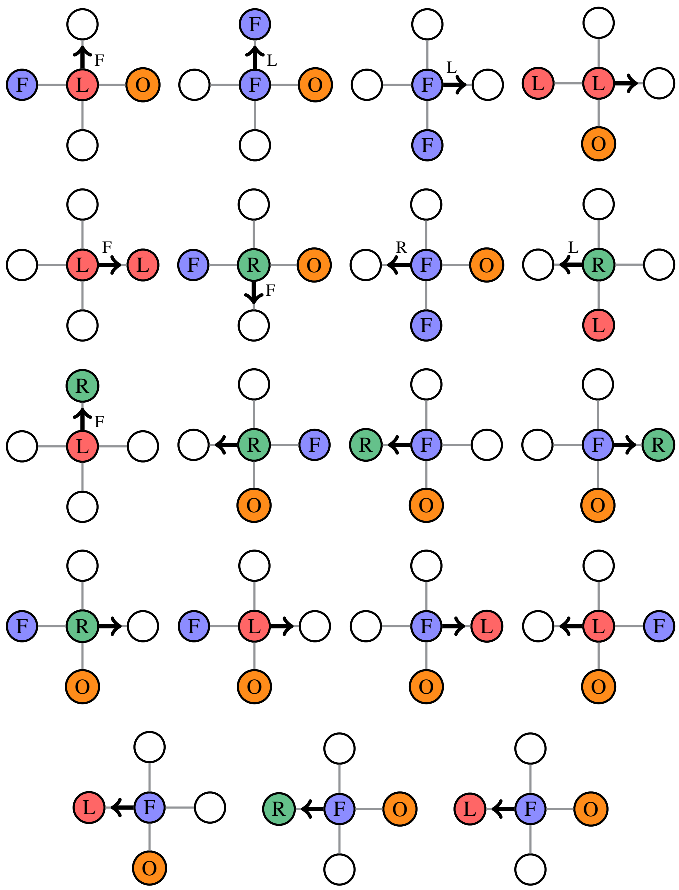


B. *Additional Rules for Generated Algorithm 1*

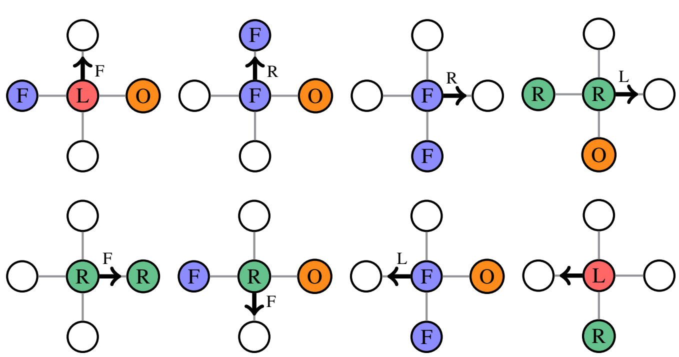


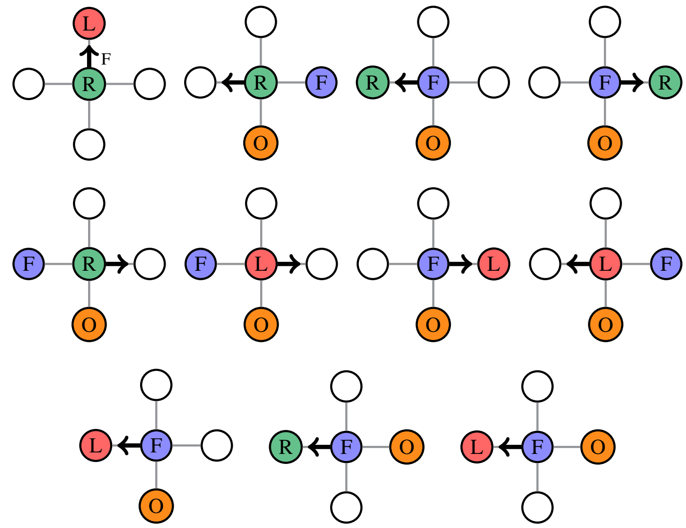


C. Additional Rules for Generated Algorithm 2

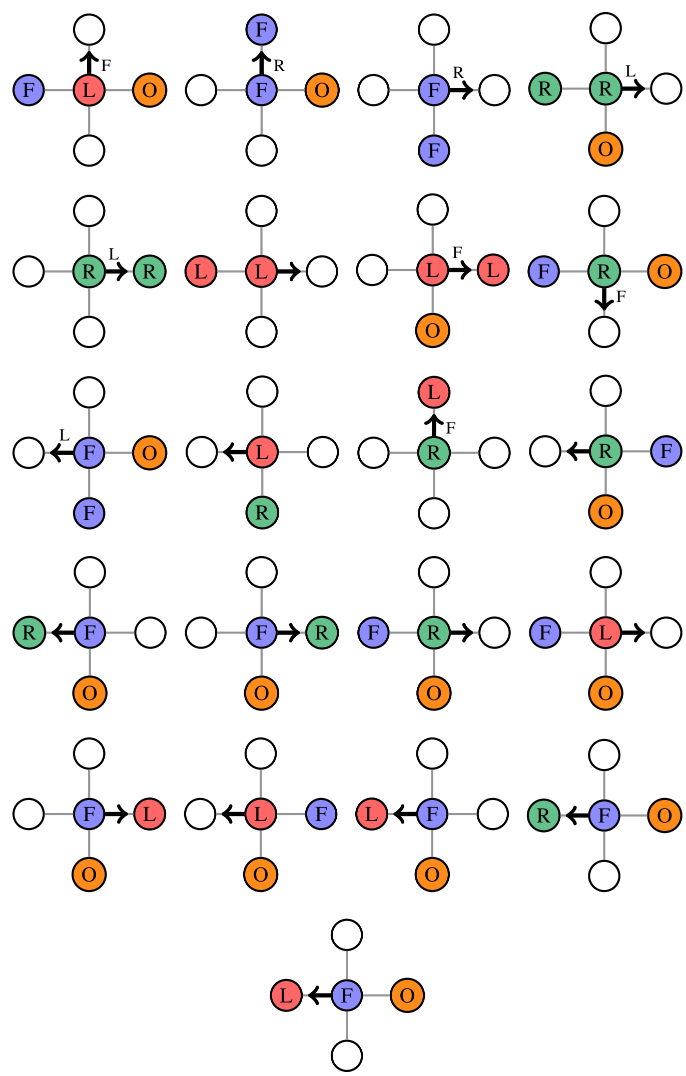


D. Additional Rules for Generated Algorithm 3

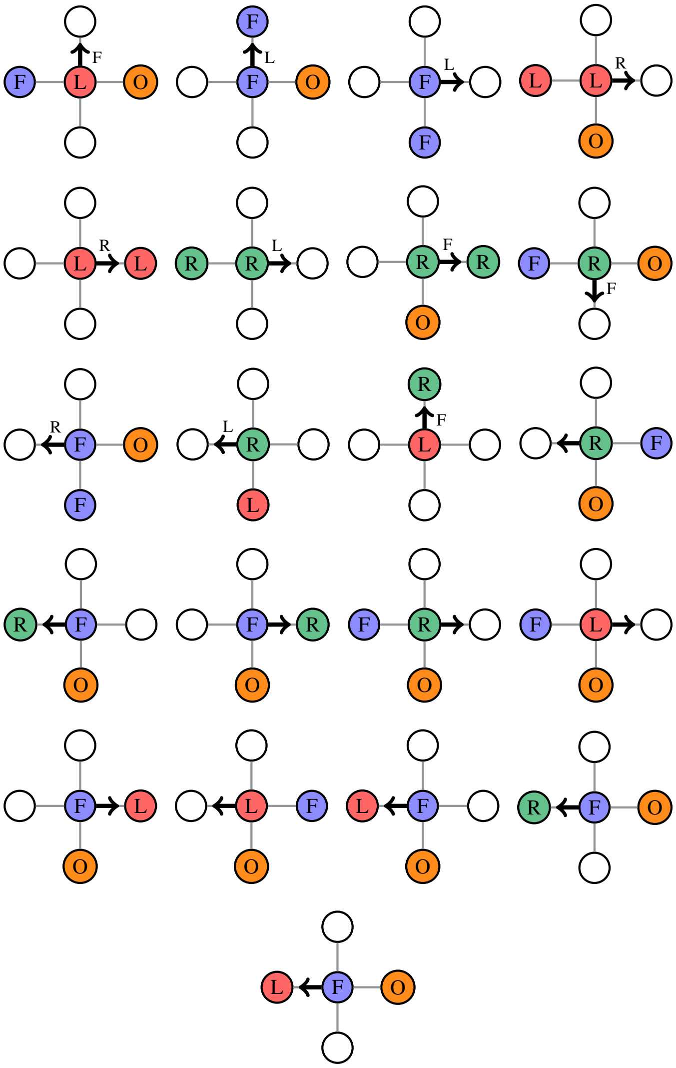




E. Additional Rules for Generated Algorithm 4



F. Additional Rules for Generated Algorithm 5



REFERENCES

[1] Q. Bramas, P. Lafourcade, and S. Devismes, “Optimal exclusive perpetual grid exploration by luminous myopic opaque robots with common chirality,” in *ICDCN*, 2021, pp. 76–85.