Collectives of compassless automata on geometric environment

Sergey Sapunov

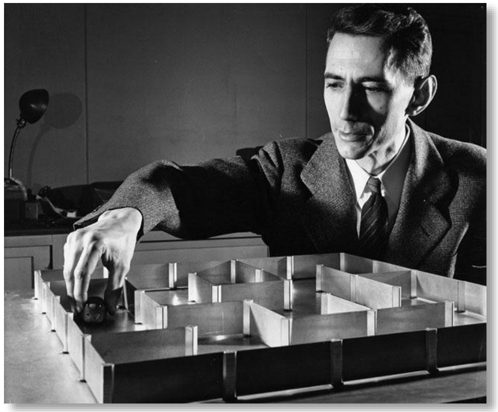
Institute of Applied Mathematics and Mechanics of the NAS of Ukraine

ALLEA
ALL European Academies

Supported by the Grant EFDS-FL2-08 of the found The European Federation of Academies of Sciences and Humanities (ALLEA)

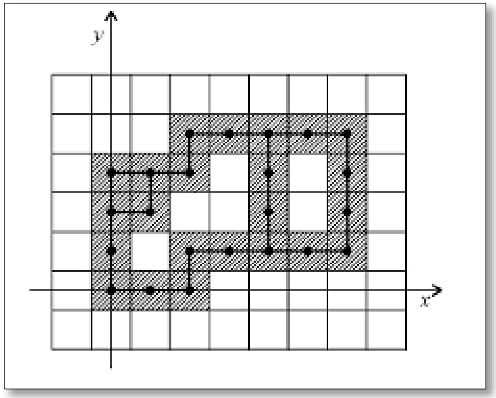
Motivation

Automata walking on graphs are a mathematical formalization of autonomous mobile agents with limited memory operating in discrete environments.



Studies on the behavior of automata in finite and infinite **labyrinths**, which are embedded directed graphs of a specific form, have emerged and are intensively developing under this model.

Hemmerling, A. (2013). Labyrinth problems: Labyrinth-searching abilities of automata (Vol. 114). Springer-Verlag.



Many **models of computation** can be represented as graph-walking automata. The graph represents the **memory** of a machine: nodes are **memory configurations**, and edges are **elementary operations** on the memory. This graph is implicitly constructed for every input object, and then the computation of the machine is interpreted as a **walk over** this graph.

Application

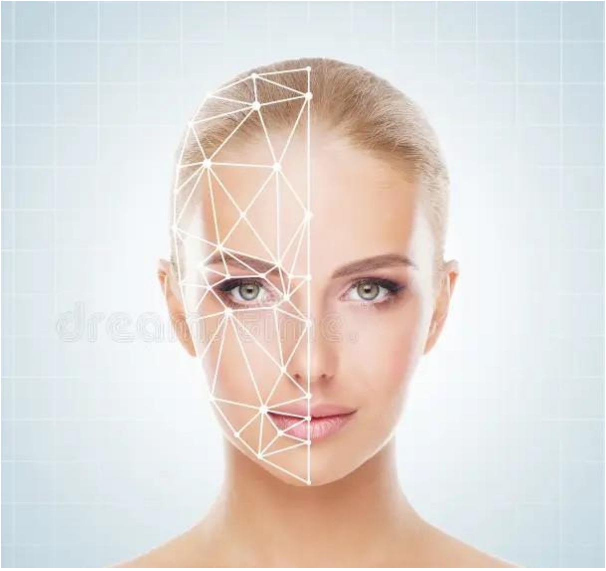


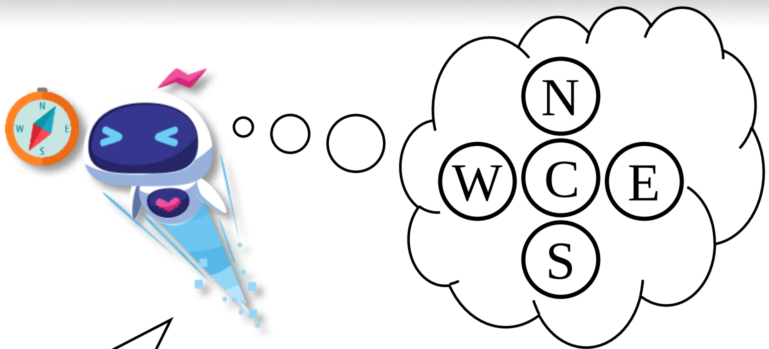
image analysis

Stamatovic, Biljana & Kilibarda, Goran. (2017). Algorithm for Identification of Infinite Clusters Based on Minimal Finite Automaton. *Mathematical Problems in Engineering*. 2017. 1-7. doi: 10.1155/2017/8251305.

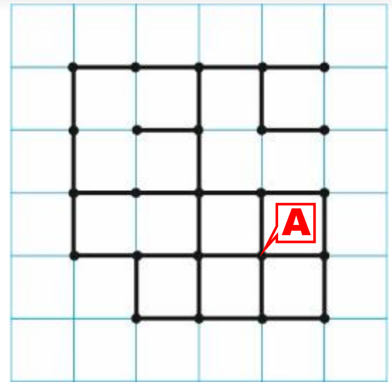
mobile robotics navigation

Dudek, G., & Jenkin, M. (2010). *Computational Principles of Mobile Robotics* (2nd ed.). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511780929

Compass

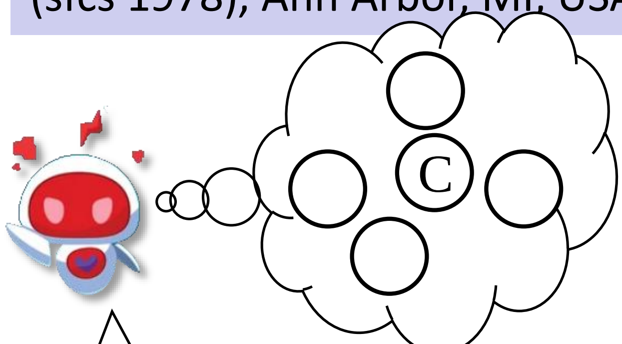


Automaton with Compass

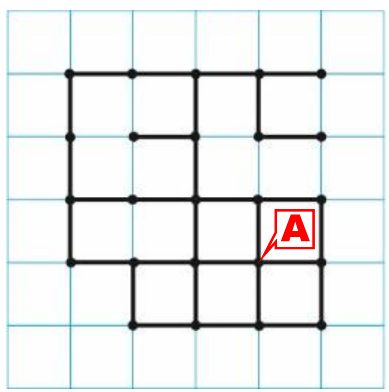


Automaton can easily maintain its movement direction.

Blum, M., & Kozen, D, (1978). On the power of the compass (or, why mazes are easier to search than graphs). In: 19th Annual Symposium on Foundations of Computer Science (sfcs 1978), Ann Arbor, MI, USA, pp. 132-142.



Compassless Automaton



Automaton need **additional equipment** and development of **methods** for its use.

Environment

Let $\mathbb{Z}$ denote the set of integers, and let $\mathbb{N}$ denote the the set of natural numbers (along with zero).

$\mathbb{Z}_m$ denote the set $\{0,1,\dots,m-1\}$ for any $m \in \mathbb{N}$.

An arbitrary set $E \subseteq \mathbb{Z}^n$ is called an n-dimensional **geometric environment**.

The elements of the set E are called the **vertices** of the environment.

Two vertices $v = (a_1,\dots,a_n)$ and $v' = (b_1,\dots,b_n)$ are called **neighbours** if $|a_i - b_i| = 1$ for only one $i \in \{1,\dots,n\}$ and $a_i = b_i$ for the remaining i.

The set of all vertices neighbouring $v \in E$ is called the **neighbourhood** of v.

The **direction** in the environment $\mathbb{Z}^n$ relative to the current vertex is any tuple of n setwise coprime integers.

We say that a vertex $v' = (b_1,\dots,b_n)$ is in the direction $(d_1,\dots,d_n)$ from a vertex $v = (a_1,\dots,a_n)$ if $b_1 = a_1 + kd_1$, …, $b_n = a_n + kd_n$, where $k \in \mathbb{N}$.

The geometric environment $E = \mathbb{Z} \times \mathbb{Z}_2$ is called a **square lattice of width 2**.

Automata in the Environment

Suppose a finite automaton A moves in environment E.

Its **input** is information about current vertex and neighbourhood.

The automaton's **output** is to move to a vertex neighbouring the current one, chosen on the basis of the input analysis.

If an automaton A distinguishes between vertices in the current neighbourhood by the coordinates of directions in the environment E, we call it the **automaton with a compass**.

Otherwise, if it does not use the coordinate system, we call it a **compassless automaton**.

Collectives of Automata

We will consider a **collective of automata** $\mathcal{A} = (A_1, \ldots, A_m)$ interacting in the environment E.

Each automaton A_i receives information about the current vertex and its neighbourhood along with information about the presence of other automata of the collective $\mathcal{A}$. $\mathcal{A}$ is said to be a **collective of compassless automata** if every automaton in $\mathcal{A}$ is a compassless automaton.

Let $J \subset \{1, \ldots, m\}$. A subsystem $\left(A_j\right)_{j \in J}$ of a collective $\mathcal{A}$ is called automata-pebbles (or **pebbles**) in this collective if for all $j \in J$ the following conditions hold: (1) A_j has a single internal state; (2) A_j can only move if there is an automaton A_i, $i \notin J$, on the same vertex, and A_j can only move to the same vertex as A_i.

The **collective of type $(\mathbf{1}, \boldsymbol{m})$** is called the collective $\mathcal{A} = (A_1, A_2, \ldots, A_{m+1})$ consisting of an automaton A_1 and m pebbles A_2, …, A_{m+1}.

Let M denote the set of all members of the collective $\mathcal{A}$.

Automaton

Automaton $A_i \in M, 1 \leq i \leq m+1$, is a sextuple

$$A_i = \left(Q_i, X_i, Y_i, q_0^i, \varphi_i, \psi_i\right)$$

where

- Q_i is a finite set of internal states;
- $X_i = \{(\alpha, \{\beta, \gamma, \delta\}) | \alpha, \beta, \gamma, \delta \subseteq M\}$ is a finite input alphabet;

Here α describes automata in the current vertex, and $\{\beta, \gamma, \delta\}$ describes automata in its neighbourhood

- $Y_i = \mathcal{P}(M) \cup \{\text{stay}\}$ is a finite output alphabet;

Here $y = \{\text{stay}\}$ means “stay at the current vertex”, $y = \emptyset$ means “move to any vertex from the current vertex’s neighbourhood that has no automata”, $y \in Y_i \setminus \{\emptyset, \text{stay}\}$ means ”move to the vertex containing only automata from the set y”

- $q_i^0 \in Q_i$ is an initial state;
- $\varphi_i: Q_i \times X_i \to Q_i$ is a a transition function;
- $\psi_i: Q_i \times X_i \to Y_i$ is an output function.

Pebble

For any **pebble** $A_j \in M, 2 \leq j \leq m+1$, the following conditions are true:

1) $Q_j = \left\{q_0^j\right\}$;

2) for any $x = (\alpha, \{\beta, \gamma, \delta\}) \in X_j$ either $\varphi_j\left(q_0^j, x\right) = \text{stay}$, or if $\varphi_j\left(q_0^j, x\right) = y \neq \text{stay}$ then $A_1 \in \alpha$ and there exists $q \in Q_1$ such that $\varphi_1(q, x) = y$.

We assume that the automaton can distinguish between all pebbles.

Behaviour

The **behaviour** (nondeterministic) of a collective $\mathcal{A} = (A_1, A_2, \dots, A_{m+1})$ of the type $(1, m)$ on the geometric environment E is the set $\Pi(\mathcal{A}, E)$ of sequences

$$\pi(\mathcal{A}, E): (\vec{x}_0, \vec{q}_0, \vec{y}_0), \dots, (\vec{x}_t, \vec{q}_t, \vec{y}_t), (\vec{x}_{t+1}, \vec{q}_{t+1}, \vec{y}_{t+1}), \dots,$$

where

$\vec{x}_t = (x_t^1, \dots, x_t^{m+1}),\, x_t^j = \left(\alpha_t^j, \left\{\beta_t^j, \gamma_t^j, \delta_t^j\right\}\right) \in X_j,$

$\vec{q}_t = (q_t^1, \dots, q_t^{m+1}),\, q_t^j \in Q_j,$

$\vec{y}_t = (y_t^1, \dots, y_t^{m+1}),\, y_t^j \in Y_j,\, 1 \le j \le m+1,$

such that $q_{t+1}^j = \varphi_j(q_t^j, x_t^j),\, y_{t+1}^j = \psi_j(q_t^j, x_t^j)$.

A single sequence $\pi(\mathcal{A}, E)$ is called an **implementation** of behaviour $\Pi(\mathcal{A}, E)$.

Movement

Let the automaton $A_i \in M$, $1 \leq i \leq m+1$, be at vertex $v_i(t) = \left(a_{1_i}, a_{2_i}\right)$ at time t. The vector

$$v_{\mathcal{A}}(t) = \frac{v_1(t) + \cdots + v_{m+1}(t)}{m+1}$$

is called the **coordinate** of collective $\mathcal{A}$ at time t.

We will call $d_{\mathcal{A}} = \max\left\{\left|a_{i_k} - a_{i_l}\right| \middle| 1 \leq i \leq 2, 1 \leq k, l \leq m+1\right\}$ the **diameter** of this collective $\mathcal{A}$.

The movement of the collective $\mathcal{A}$ in the environment E during the implementation of behavior $\pi(\mathcal{A}, E)$ is called **directed** if there are constants $c_1, c_2 \in \mathbb{N}$ such that the diameter $d_{\mathcal{A}} \leq c_1$ and for any time t there are natural $t', t'' \leq c_2$ for which the equality $v_{\mathcal{A}}(t+t') - v_{\mathcal{A}}(t) = v_{\mathcal{A}}(t+t'+t'') - v_{\mathcal{A}}(t+t')$ holds.

We assume that the movement of a collective $\mathcal{A}$ in the environment E is directed if it is directed for all implementations of its behaviour $\Pi(\mathcal{A}, E)$. Otherwise, we say that the movement of this collective is not directed.

Pebbles Configuration

Pebble configuration will be considered as their relative position on the lattice vertices.

The initial position of the pebbles is called their **initial configuration.**

We will say that the configurations K' and K'' of the pebbles of collective $\mathcal{A}$ are **indistinguishable** if any implementation of the behaviour of this collective with the initial configuration K' is the implementation of the behaviour of the same collective with the initial configuration K'' and vice versa.

We say that two configurations of the pebbles of collective $\mathcal{A}$ are **indistinguishable in the worst case** if there is at least one implementation of the behaviour of this collective that is the same in both configurations.

Schema of Configuration

A **schema $\widetilde{K}$ of a configuration** of pebbles K is a fragment of the environment, where the vertices on which the pebbles are located are marked in some way.

Different configurations can have the same schema.

Let's say that each specific configuration of pebbles is an **interpretation** of the corresponding schema.

Schemes $\widetilde{K}'$ and $\widetilde{K}''$ are said to be **indistinguishable** if for any interpretation of scheme $\widetilde{K}'$ there is an indistinguishable interpretation of scheme $\widetilde{K}''$ vice versa.

We define the **worst-case indistinguishability** of schemas in the same way as we did for configurations.

An **elementary transfer of a pebble** is the relocation of one pebble from its current vertex to a neighbouring vertex. The movements of the pebbles of collective $\mathcal{A}$ can be represented as a sequence of their elementary transfers and, therefore, as a sequence of configurations or schemes.

Sub-Collective

Let **sub-collective** $\mathcal{A}'$ of collective $\mathcal{A}$ be an arbitrary subset of the members of that collective.

We say that sub-collective $\mathcal{A}'$ is **isolated** in collective $\mathcal{A}$ if there exists an implementation of the behaviour of that collective in which no member of $\mathcal{A}'$ observes any member of $\mathcal{A}$ not belonging to $\mathcal{A}'$.

It is obvious that from the isolation of some sub-collective $\mathcal{A}'$ of collective $\mathcal{A}$ follows the isolation of the sub-collective $\mathcal{A} \setminus \mathcal{A}'$.

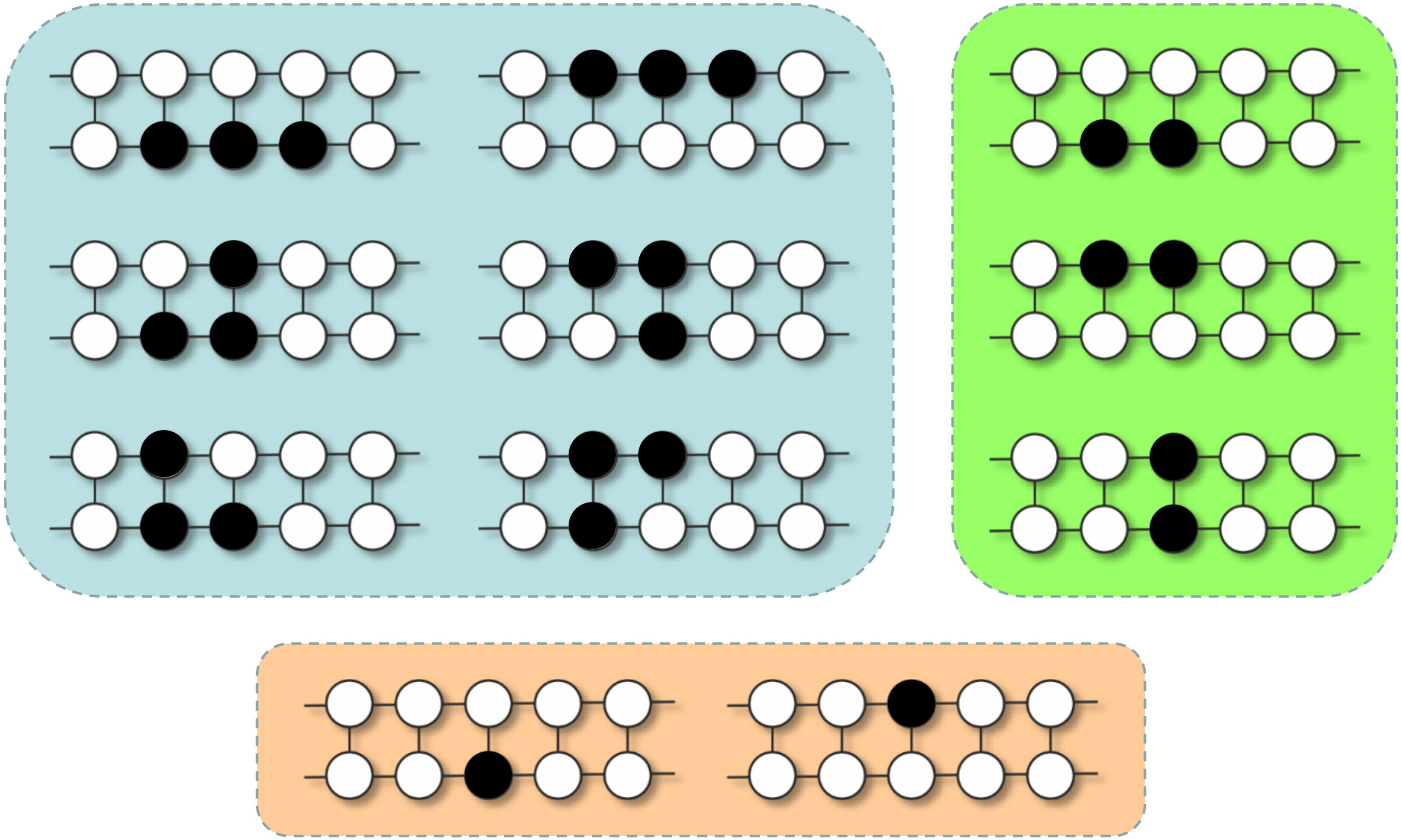
Negative Propositions

Теорема 1

The movement of the following collectives on a square lattice of width 2 is not directed:

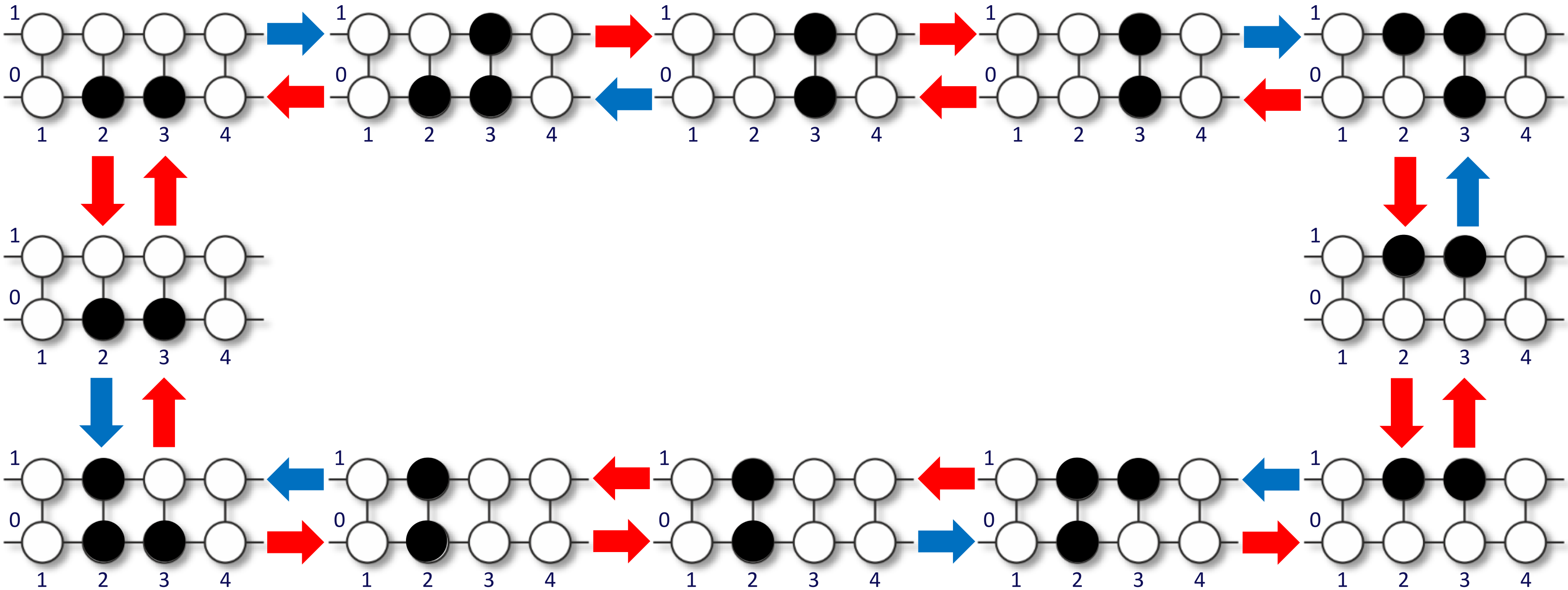
1. *a collective of a single automaton*;
2. *a collective of an automaton and a pebble*;
3. *a collective of one automaton and two pebbles*;
4. *a collective of one automaton and three pebbles*.

Type (1, 3) – Schemes



Sets consist of schemes pairwise indistinguishable in the worst case.

Type (1, 3) – Example



The example shows a diagram of the transformation of schemes in which the pebbles in the worst case do not leave some limited area of the lattice.

Red arrow denotes the transfer of a pebble to an occupied vertex.

Blue arrow represents the transfer to a free vertex (**nondeterministic**).

Positive Proposition

Теорема 2

There exists a collective of one automaton and four pebbles that perform a directed movement on a square lattice of width 2.

Let us construct a collective $\mathcal{A} = (A_1, A_2, A_3, A_4, A_5)$ that performs directed movement in the environment E.

Let us define variables B, C, D, and H which have values in the set of pebbles (A_2, A_3, A_4, A_5), where $B \neq C \neq D \neq H$.

Assume that the direction in which the collective $\mathcal{A}$ should move is determined by the initial position of the pebbles B, C and D.

Algorithm – Pseudocode

Algorithm 1 Directed movement of collective $\mathcal{A}$ of type (1,4)

Require: initial coordinate
Ensure: a sequence of coordinates that satisfies the definition of directed movement

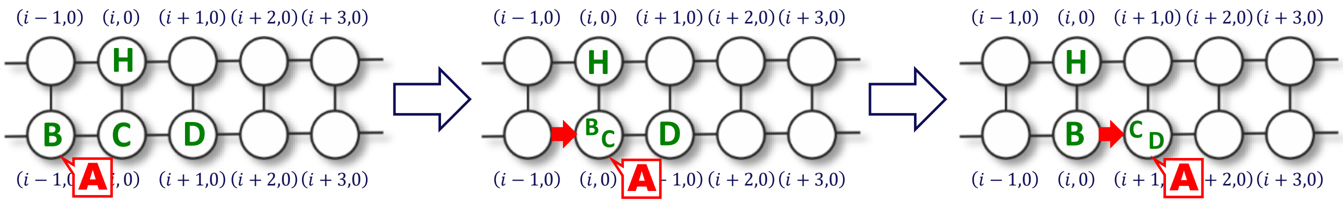
loop
  A_1 and B move to the vertex where C is located
  A_1 and C move to the vertex where D is located
  A_1 and D move to a free vertex
  if there is H in the neighbourhood of the current vertex **th**
    A_1 moves to the vertex where H is located
    A_1 and H move to the vertex where D is located
    A_1 and D move to the vertex where C is located
    A_1 and D move to a free vertex

Algorithm 1 Directed movement of collective $\mathcal{A}$ of type (1,4)

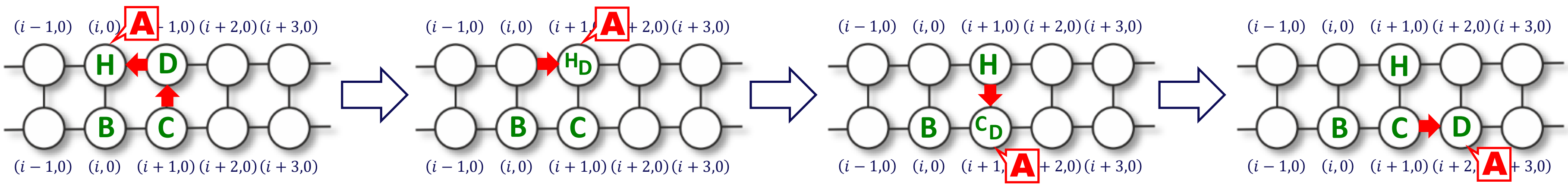
  else
    A_1 moves to the vertex where C is located
    A_1 moves to the vertex where B is located
    A_1 moves to the vertex where H is located
    A_1 and H move to the vertex where B is located
    A_1 and H move to the vertex where C is located
    A_1 and H move to a free vertex
  end if
  A_1 moves to the vertex where C is located
  A_1 moves to the vertex where B is located
end loop

Algorithm – Execution

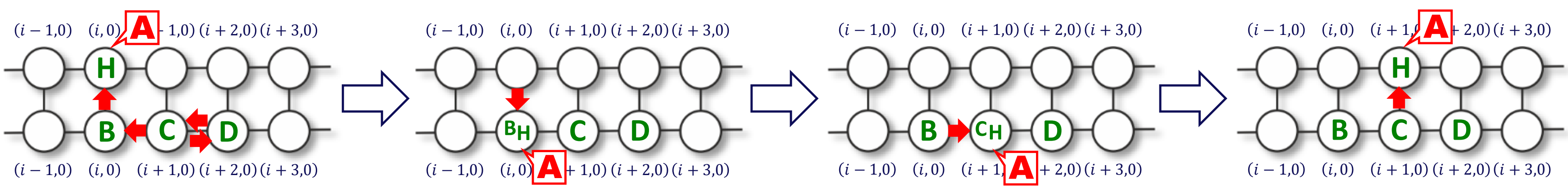
Common Part (lines 2–3)



Branch 1 (lines 4, 6–9)



Branch 2 (lines 4, 11–16)



Algorithm – Correctness

Initially. $t = 0$. $\mathcal{A}$ at $(i-1,0)$, $(i,0)$, $(i+1,0)$, $(i,1)$; $d_{\mathcal{A}} = 2$; $v_{\mathcal{A}} = \left(i - \frac{1}{5}, \frac{1}{5}\right)$

After 1st iteration. $t' \in \{9, 11\}$ indicate the duration of the iteration.

$\mathcal{A}$ at $(i,0)$, $(i+1{,}0)$, $(i+2,0)$, $(i+1{,}1)$; $d_{\mathcal{A}} = 2$; $v_{\mathcal{A}} = \left(i + \frac{1}{5}, \frac{1}{5}\right)$

After 2nd iteration. $t'' \in \{9, 11\}$ indicate the duration of the iteration.

$\mathcal{A}$ at $(i+1,0)$, $(i+2{,}0)$, $(i+3,0)$, $(i+2{,}1)$; $d_{\mathcal{A}} = 2$; $v_{\mathcal{A}} = \left(i + \frac{9}{5}, \frac{1}{5}\right)$

$$v_{\mathcal{A}}(t') - v_{\mathcal{A}}(0) = v_{\mathcal{A}}(t' + t'') - v_{\mathcal{A}}(t') = (1, 0)$$

After k-th iteration. $t \in \{9, 11\}$ indicate the duration of the iteration.

$\mathcal{A}$ at $(i+k-1,0)$, $(i+k,0)$, $(i+k+1,0)$, $(i+k,1)$; $d_{\mathcal{A}} = 2$; $v_{\mathcal{A}} = \left(i + k + \frac{1}{5}, \frac{1}{5}\right)$

After $(k+1)$-th iteration. $t' \in \{9, 11\}$ indicate the duration of the iteration.

$\mathcal{A}$ at $(i+k,0)$, $(i+k+1{,}0)$, $(i+k+2,0)$, $(i+k+1{,}1)$; $d_{\mathcal{A}} = 2$; $v_{\mathcal{A}} = \left(i + k + \frac{4}{5}, \frac{1}{5}\right)$

After $(k+2)$-th iteration. $t'' \in \{9, 11\}$ indicate the duration of the iteration.

$\mathcal{A}$ at $(i+k+1,0)$, $(i+k+2{,}0)$, $(i+k+3,0)$, $(i+k+2{,}1)$; $d_{\mathcal{A}} = 2$; $v_{\mathcal{A}} = \left(i + k + \frac{9}{5}, \frac{1}{5}\right)$

$$v_{\mathcal{A}}(t' + t'') - v_{\mathcal{A}}(t) = v_{\mathcal{A}}(t + t' + t'') - v_{\mathcal{A}}(t + t') = (1, 0)$$

Publications

Sapunov, S. (2023). Directional movement of a collective of compassless automata on square lattice of width 2. *arXiv preprint arXiv:2309.00292*.

Submitted to *Cybernetics and Systems Analysis*

Thank you!