

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 13(3), 303-320, June 2018.

On Distributed Solution to SAT by Membrane Computing

H.N. Adorna, L. Pan, B. Song

Henry N. Adorna

Algorithms and Complexity Lab.
Department of Computer Science
University of the Philippines Diliman
Diliman 1101 Quezon City, Philippines
hnadorna@up.edu.ph

Linqiang Pan*

1. Key Laboratory of Image Information Processing and
Intelligent Control of Education Ministry of China
School of Automation
Huazhong University of Science and Technology
Wuhan 430074, Hubei, China
2. School of Electrical and Information Engineering
Zhengzhou University of Light Industry
Zhengzhou 450002, Henan, China
*Corresponding author: lqpan@mail.hust.edu.cn

Bosheng Song

Key Laboratory of Image Information Processing and
Intelligent Control of Education Ministry of China
School of Automation
Huazhong University of Science and Technology
Wuhan 430074, Hubei, China
boshengsong@hust.edu.cn

Abstract: Tissue P systems with evolutionary communication rules and cell division (TPec, for short) are a class of bio-inspired parallel computational models, which can solve NP-complete problems in a feasible time. In this work, a variant of TPec, called k -distributed tissue P systems with evolutionary communication and cell division (k - Δ_{TPec} , for short) is proposed. A uniform solution to the SAT problem by k - Δ_{TPec} under balanced fixed-partition is presented. The solution provides not only the precise satisfying truth assignments for all Boolean formulas, but also a precise amount of possible such satisfying truth assignments. It is shown that the communication resource for one-way and two-way uniform k -P protocols are increased with respect to k ; while a single communication is shown to be possible for bi-directional uniform k -P protocols for any k . We further show that if the number of clauses is at least equal to the square of the number of variables of the given boolean formula, then k - Δ_{TPec} for solving the SAT problem are more efficient than TPec as show in [39]; if the number of clauses is equal to the number of variables, then k - Δ_{TPec} for solving the SAT problem work no much faster than TPec.

Keywords: Membrane computing, distributed P system, SAT, communication complexity

1 Introduction

Since the research area of membrane computing was proposed in 1998 [8, 19], the research lines about computation power of various variants of P systems [25, 26, 26, 33, 37], their applications [8, 24, 29, 36] and implementation issues [15, 16, 32] have been investigated widely. Several solution

approaches and techniques have been presented in the literature for solving Satisfiability Problem (SAT) using many variants of P systems [11, 12, 27, 30]. Each of these variants of P systems solving SAT provided better solutions than the conventional model in terms of time efficiency or computational time complexity [18, 31, 38]. Most of them are benefited from the nondeterministic maximal parallelism of P systems and its ability to produce exponentially many cells or regions (in linear time) in a computation [35].

Evolution and communication are the core operations in the solutions offered by the variants of P systems, where communication allows objects to be transmitted to the other regions/cells for further processing. In [14], Hernandez, et al. provided a solution to 3-SAT, where the amount of communications is measured from a dynamic communication complexity perspective [1]. Several results might have been reported using communication as a measure of complexity [2, 5, 6], but the analyses would be dissimilar to that of [1].

In order to capture the concept of communication complexity as introduced by A.C. Yao in [34], Gh. Păun, et al. [21] introduced and defined a so-called *dP scheme*, where the input of a *k-dP* scheme is partitioned and these parts are distributed to the participating components. Necessarily, these components need to communicate to solve a problem, and the so-called *inter-component communication rules* are introduced as a new kind of set of communication rules.

Computation done by distributed model in this work depends on the agreed upon partition of the input among the participating P systems. A partition is called *balanced partition* if the number of objects or the length of the part of the input assigned to each participating component P systems is almost equal. Let w be an input for a distributed system with two components. The partition w_1 and w_2 of w , where w_i is assigned to Π_i , $i = 1, 2$ is *balanced* if and only if $||w_1| - |w_2|| \leq 1$. In particular, it is a *balanced fixed-partition*, if $w = w_1w_2$; otherwise, we have *unbalanced (fixed-)partition*.

The distributed P systems halt if and only if all participating component P systems halt. If a distributed system halts in a specified accepting configuration, then it accepts/decides/solves a problem.

Since *k-dP* scheme was introduced, some of the results on this variant were reported in [7, 10, 23]. In [4], Buño, et al. introduced a distributed solution for *n*-queens problem as presented in Naranjo, et al. [13]. Indeed, in [4], the reduction of the *n*-queens problem to SAT of a *dP* scheme were used, where the components are P systems with active membranes.

Recently, Buño, et al. [3] capitalized the power of tissue P systems (*TPec*) with evolutionary communication rules and cell division introduced in [39], in proposing a distributed solution to SAT. In particular, they defined a so-called *2-dTPec*. The paper claimed a decent advantage of distributed solution compared to the non-distributed one with respect to the results reported in [39]. Also, only two participating components in systems were considered working on a balanced partition.

In this work, the solution presented in [3] will be revisited and some other insights into the consequences of our results are provided. Some other relevant issues related to communication complexity in distributed tissue P systems with evolutionary communication and cell division are proposed.

Contributions of the present work are summarized as follows:

- (a) A variant of tissue P systems with evolutionary communication and cell division, called *k*-distributed tissue P systems with evolutionary communication and cell division (*k-ΔTPec*, for short) and the corresponding recognizer version are proposed.
- (b) A uniform solution to the SAT problem by *k-ΔTPec* under balanced fixed-partition is presented. The solution provides not only the precise satisfying truth assignments for all Boolean formulas, but also a precise amount of possible such satisfying truth assignments.

- (c) The communication resource for one-way and two-way uniform k -P protocols is increased with respect to k ; while a single communication is shown to be possible for bi-directional uniform k -P protocols for any k .
- (d) We further show that if the number of clauses is at least equal to the square of the number of variables of the given boolean formula, then k - Δ_{TPec} for solving the SAT problem are more efficient than TPec as show in [39]; if the number of clauses is equal to the number of variables, then k - Δ_{TPec} for solving the SAT problem works no much faster than TPec.

The rest of this work is organized as follows. Section 2 provides the preliminaries of a dP scheme, introducing the concepts of P protocols and balanced (fixed) partition and the communication resources used in the analyses of the solution to SAT. In Section 3, distributed tissue P systems with evolutional communication and cell division or k - Δ_{TPec} are defined. Solution to SAT using 2 - Δ_{TPec} is presented in Section 3, while solution to SAT using 3 - Δ_{TPec} is given in Section 5. Remarks on the relative efficiency of distributed solutions are provided in Section 4. Finally, conclusions and discussions are given in Section 5.

2 Preliminaries

In this section, the notions of k -dP scheme and communication complexity of P systems are presented, then tissue P systems with cell division and evolutional communication rules are introduced [39].

Definition 1. A k -dP scheme ($k \geq 2$) is a tuple

$$k\text{-}\Delta_{\Pi} = (\Gamma, 0, \Pi_1, \Pi_2, \dots, \Pi_k, R_{\Delta}),$$

where

- Π is a fixed variant of P system;
- Γ is an alphabet of objects in the whole system Δ ;
- 0 is the common/shared environment of Π_i ;
- Π_i for $i = 1, 2, \dots, k$ are P systems of the fixed variant Π with Γ as working alphabet, skin membranes or local environments of each P system will be labelled injectively as s_1, s_2, \dots, s_k ;
- R_{Δ} is a finite set of rules of the form $(s_i, u/v, s_j)$, where $1 \leq i, j \leq k$, $i \neq j$, and $u, v \in \Gamma^*$, such that $uv \neq \lambda$. We denote by $|uv|$ the *weight* of the rule $(s_i, u/v, s_j)$. This antiport-like communication rule is called inter-component communication rule.

The mechanism by which a k -dP scheme performs its computation could be found in [21]. In particular, an input for a k -dP scheme is partitioned into k parts and distributed one part to each of the k components of the dP scheme. Thus, communication to solve the problem is inevitable. In this paper, definition of *balanced* and *unbalanced* partition of an input is provided.

Definition 2. A partition $\{P_1, P_2, \dots, P_k\}$ is called balanced partition if and only if for all i , P_i have the same size or at most a difference of 1. Otherwise, we call it an unbalanced partition.

Definition 3. We call a k -balanced (unbalanced) partition $\{P_1, P_2, \dots, P_k\}$ (resp., a k -balanced (unbalanced) fixed-partition) if and only if the k -partition of input is done from left to right with respect to the (resp., fixed) ordering of the input.

Cooperation between component P systems of a dP scheme is defined in the set R_Δ of inter-component communications. R_Δ specifies the mode of communication protocol of a dP scheme. In what follows, k -P protocols for a k -dP scheme are defined.

Definition 4. Let $k\text{-}\Delta_\Pi$ be a k -dP scheme.

- $k\text{-}\Delta_\Pi$ is called 1-way k -P protocol if and only if R contains only rules of the form $(s_i, u/\lambda, s_j)$.
- $k\text{-}\Delta_\Pi$ is called 2-way k -P protocol if and only if R contains rules of the form $(s_i, u/\lambda, s_j)$ and $(s_i, \lambda/v, s_j)$.
- $k\text{-}\Delta_\Pi$ is called bi-directional k -P protocol if and only if R contains rules of the form $(s_i, u/v, s_j)$, $(s_i, u/\lambda, s_j)$ and $(s_i, \lambda/v, s_j)$.

A k -dP scheme computes as follows. All component P systems of a k -dP scheme are aware of the problem that they are solving. Each component P system knows only the part of the input assigned to them. We allow each component P system to perform computation to the input part known to them. To solve the problem, component P systems must communicate with respect to a particular protocol.

Definition 5. A configuration δ_j of $k\text{-}\Delta_\Pi$ is a vector in $\mathcal{C} = 0 \times \mathcal{M}_{\langle 1,0 \rangle} \times \mathcal{M}_{\langle 2,0 \rangle} \times \dots \times \mathcal{M}_{\langle k,0 \rangle}$, where 0 is the common environment, and $\mathcal{M}_{\langle i,0 \rangle}$ are sets of multisets of objects in $\langle i,0 \rangle$, $i = 1, 2, \dots, k$. In particular, $\delta_j = (m_{0j}, m_{1j}, m_{2j}, \dots, m_{kj}) \in \mathcal{C}$ indicates that at time j , $m_0 \in 0$ and $m_i \in \mathcal{M}_{\langle i,0 \rangle}$, $i = 1, 2, \dots, k$.

Definition 6. A computation of $k\text{-}\Delta_\Pi$ is the transition of configurations represented by a sequence $\delta : \delta_0 \Rightarrow \delta_1 \Rightarrow \dots \Rightarrow \delta_h$, where δ_0 is the initial configuration, and δ_h is the final or halting configuration. The initial configuration δ_0 is a vector of initial multisets contained in 0 (local environment of component P systems). δ is a halting computation if and only if δ_h is a configuration, where one of the objects **yes** or **no** is contained in some (specified) membrane of the system.

$k\text{-}\Delta_\Pi$ has an accepting configuration if and only if δ is a halting computation and at configuration δ_h , object **yes** appeared in a specified membrane in the system. δ is a rejecting computation if and only if at δ_h object **no** appeared.

Definition 7. A language L is decided by $k\text{-}\Delta_\Pi$, $L = L(k\text{-}\Delta_\Pi)$ if and only if for every input w over some alphabet, there is a halting computation δ of $k\text{-}\Delta_\Pi$ that decides on $w \in L$.

In this work, the existence of at least an object **yes** in 0 implies an affirmative decision on a problem, while the appearance of at least an object **no** in 0 connotes a negative decision.

All component P systems with the same and uniform procedure in processing input part, that is to say, component P systems of a k -dP scheme would be all the same, is called a *uniform k -dP scheme*. If we have a uniform $k\text{-}\Delta_\Pi$, then k -P protocol is called *uniform k -P protocol*.

This work focuses on the amount of communications used by component P systems in deciding the satisfiability of some formula φ in conjunctive normal form (CNF). Thus, the following notions are used [1].

Definition 8. [21] Let Δ be a dP scheme, $\delta : \delta_0 \Rightarrow \delta_1 \Rightarrow \dots \Rightarrow \delta_h$ is a halting computation in Δ , where δ_0 is the initial configuration. Then for each $i = 0, 1, \dots, h - 1$, we have the following parameters:

- $ComN(\delta_i \Rightarrow \delta_{i+1}) = \begin{cases} 1, & \text{if a communication rule is used in this transition,} \\ 0, & \text{otherwise,} \end{cases}$

- $ComR(\delta_i \Rightarrow \delta_{i+1})$ denotes the number of communication rules used in this transition,
- $ComW(\delta_i \Rightarrow \delta_{i+1})$ denotes the total weight of the communication rules used in this transition.

The above mentioned parameters can also be used to measure computations, results of computations, systems and languages (problems).

Definition 9. Let $L(\Delta)$ be the set of strings accepted by Δ . For $X \in \{N, R, W\}$, we define:
 $ComX(\delta) = \sum_{i=0}^{h-1} ComX(\delta_i \Rightarrow \delta_{i+1})$, δ is a halting computation,
 $ComX(w, \Delta) = \min\{ComX(\delta) \mid \delta \text{ is an accepting computation of } \Delta \text{ for } w\}$,
 $ComX(\Delta) = \max\{ComX(w, \Delta) \mid w \in L(\Delta)\}$.

3 Distributed recognizer tissue P systems

In this section, we introduce the notions of (recognizer) tissue P systems with evolutionary symport/antiport rules and cell division and k -distributed tissue P systems with evolutionary symport/antiport rules and cell division.

Definition 10. A tissue P system (of degree $q \geq 1$) with evolutionary symport/antiport rules and cell division (TPec) is a tuple

$$\Pi = (\Gamma, \mathcal{E}, \mathcal{M}_1, \dots, \mathcal{M}_q, R, i_{out}),$$

where

1. Γ is a working alphabet of objects;
2. $\mathcal{E} \subseteq \Gamma$ is the set of objects initially located in the environment;
3. \mathcal{M}_i , $1 \leq i \leq q$, are finite multisets over Γ ;
4. R is a finite set of rules of the following forms:
 - (a) Evolutional communication rules:
 - i. Evolutional symport rules: $[u]_i []_j \rightarrow []_i [u']_j$, for $1 < i \leq q, 0 < j \leq q, i \neq j; u \in \Gamma^+, u' \in \Gamma^*$ or $i = 0, 1 < j \leq q; u \in \Gamma^+, u' \in \Gamma^*$, and there exists at least one object $a \in \Gamma \setminus \mathcal{E}$, which is in cell $i = 0$. The length of an evolutional symport rule is defined as $|u| + |u'|$.
 - ii. Evolutional antiport rule: $[u]_i [v]_j \rightarrow [v']_i [u']_j$, where $0 \leq i \neq j \leq q, u, v \in \Gamma^+, u', v' \in \Gamma^*$. The length of an evolutional antiport rule is defined as $|u| + |u'| + |v| + |v'|$.
 - (b) Division rules: $[a]_i \rightarrow [b]_i [c]_i$, where $i \in \{1, 2, \dots, q\}, i \neq i_{out}, a, b, c \in \Gamma$.
5. $i_{out} \in \{1, 2, \dots, q\}$.

The details of the mechanism on how TPec works can be found in [39]. In what follows, we introduce the notion of recognizer TPec systems.

Definition 11. A recognizer tissue P system with evolutionary symport/antiport rules and cell division of degree $q \geq 1$ is a tuple

$$\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \dots, \mathcal{M}_q, R, i_{in}, i_{out}),$$

where

- Γ has two distinguished objects **yes** and **no**;
- $\Sigma \subseteq \Gamma$ is the input alphabet;
- $\mathcal{M}_1, \dots, \mathcal{M}_q$ are finite multisets over $\Gamma \setminus \Sigma$;
- $i_{in} \in \{1, \dots, q\}$ is the label of the input cell, and $i_{out} = 0$;
- all computations halt;
- either an object **yes** or an object **no** is released into the environment at the last step of any computation.

Theorem 12. [39] *Let $PMCT_{DEC(k)}$ be the set of all decision problems solvable in a uniform way and polynomial time by means of recognizer tissue P systems with cell division and evolutionary communication rules of length at most k . Then, $SAT \in PMCT_{DEC(4)}$.*

In this work, at least two recognizer tissue P systems with evolutionary symport/antiport rules and cell division of degree $q \geq 1$ are used to solve the SAT problem, where the input multiset is partitioned with respect to the number of component recognizer tissue P systems. Consequently, a so-called *k-distributed tissue P system with evolutionary symport/antiport rules and cell division rules* (*k-dTPec system*) is defined.

Definition 13. A *k-distributed tissue P system with evolutionary symport/antiport rules and cell division* or *k-dTPec system* is defined as follows:

$$k\text{-}\Delta_{TPec} = (\Gamma, 0, \Pi_1, \Pi_2, \dots, \Pi_k, R_\Delta, i_{out}),$$

where

1. Γ is a set of alphabet of objects in the whole system Δ ;
2. 0 is the common or shared environment of all Π_i , $i = 1, 2, \dots, k$;
3. Π_i , $i = 1, 2, \dots, k$ are recognizer tissue P systems with evolutionary symport/antiport rules and cell division. Each Π_i has the alphabet of objects in Γ . Each cell of Π_i will be labelled $\langle i, j \rangle$, where $j = 1, 2, \dots, d_i$, and d_i denotes the number of cells of Π_i . The external region or the environment is different for each component. We will refer to the environment of component i as the local environment of i and is denoted by the label $\langle i, 0 \rangle$;
4. R_Δ is a set of finite inter-component communication rules of the form: $(\langle i, 0 \rangle, u/v, \langle j, 0 \rangle)$, where $\langle i, 0 \rangle$ and $\langle j, 0 \rangle$ are the local environments of components i and j , respectively, $u, v \in \Gamma^*$, and $|uv| \geq 1$;
5. i_{out} is the component of the dTPec designated as the output component. Only the objects produced in the output region of the system are considered as output in a halting computation of the dTPec.

Note that the *k-dP* tissue P system used in the next section is actually a uniform *k-distributed* recognizer tissue P system with evolutionary symport/antiport rules.

4 Solving SAT by $2\text{-}\Delta_{TPec}$

In this section, a $2\text{-}\Delta_{TPec}$ is presented that solves the satisfiability of any instance φ of the SAT problem. In particular, a uniform 2-P protocols is constructed which is based on the construction in [39].

Theorem 14. *Let φ be any instance of the SAT problem in CNF with m clauses and n variables. Then there exists $2\text{-}\Delta_{TPec}$ deciding on satisfiability of φ under a balanced fixed-partition in $3n + 3\lceil \frac{m}{2} \rceil + 4$ steps using antiport-like inter-component communication rules.*

Proof: A $2\text{-}\Delta_{TPec}$ for the SAT problem will be constructed such that component P systems are quite the same with that presented in [39] but with some additional rules.

Let $2\text{-}\Delta_{TPec}$ be defined as follows:

$$2\text{-}\Delta_{TPec} = (\Gamma, 0, \Pi_1, \Pi_2, R_\Delta),$$

where

- $\Gamma = \Gamma_1 \cup \Gamma_2$,
- 0 is the common shared environment of the Π_1 and Π_2 ,
- Π_k , $k = 1, 2$ are recognizer tissue P systems defined as:

$$\Pi_k(\langle n, m \rangle) = (\Gamma_k, \Sigma_k, \mathcal{E}_k, \mathcal{M}_{k,1}, \mathcal{M}_{k,2}, \mathcal{M}_{k,3}, \mathcal{M}_{k,x}, R_k, i_{kin} = 2, i_{kout} = 0),$$

where:

- $\Gamma_k = \Gamma'_k \cup V$,

$$\begin{aligned} \Gamma'_k &= \Sigma_h \cup \{a_i, t_i, f_i, t'_i, f'_i, \beta_i, \beta'_i \mid 1 \leq i \leq n\} \\ &\cup \{e_{i,j}, e'_{i,j}, \bar{e}_{i,j}, \bar{e}'_{i,j}, E_{i,j}, \bar{E}_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq m\} \\ &\cup \{d_{i,j,h}, d'_{i,j,h}, \bar{d}_{i,j,h}, \bar{d}'_{i,j,h} \mid 1 \leq i \leq n, 1 \leq j \leq m, 1 \leq h \leq n-1\} \\ &\cup \{b_j c_j, \bar{c}_j, \bar{c}'_j, E_j \mid 1 \leq j \leq m\} \\ &\cup \{b_{j,h}, b'_{j,h} \mid 1 \leq j \leq m, 1 \leq h \leq n-1\} \\ &\cup \{\alpha, \alpha'_i \mid 0 \leq i \leq 3n + 3m_k\} \\ &\cup \{d, E_{m_k+1}, \alpha_{3n+3m_k+1}, \text{yes}, \text{no}, y, y'\}, \end{aligned}$$
- $V = \{v_l \mid 1 \leq l \leq 2^n, \text{ where } n \text{ is the number of variables}\}.$
- $\Sigma_k = \{x_{i,j}, \bar{x}_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq m\},$
- $\mathcal{E}_k = \{\alpha'_i \mid 0 \leq i \leq 3n + 3m_k\},$
- $\mathcal{M}_{k,1} = \{a_1, \dots, a_n, E_1\}, \mathcal{M}_{k,2} = \{b_1, \dots, b_n, d, \alpha_0\}, \mathcal{M}_{k,3} = \{a_1, \dots, a_n\},$ and $\mathcal{M}_{k,x} = \emptyset,$
- R_k is the set of rules of each $k = 1, 2$ component.

The set of rules we will use are those set of rules $r_{1,i}$ until $r_{26,i}$, from [39] only with the following modifications and addition:

- (a) we split $r_{1,i}$ into $r_{1,i;c}$, where c indicates in which cell $r_{1,i}$ will be applied on variable i :

$$\begin{aligned} r_{1,i;3} &\equiv [a_1]_3 \rightarrow [\beta_1]_3 [\beta'_1]_3, \quad 1 \leq i \leq n, \\ r_{1,i;1} &\equiv [a_i]_1 \rightarrow [t'_i]_1 [f'_i]_1, \quad 1 \leq i \leq n; \end{aligned}$$

- (b) we replace r_{27} with the following:

$$r_{27} \equiv [v_l E_{m_k+1}]_1 [v_l]_3 \rightarrow [y]_1 [v_l \text{yes}]_3;$$

- (c) we replace r_{28} with the following:
 $r_{28} \equiv [v_l \mathbf{yes}]_3 []_0 \rightarrow []_3 [v_l \mathbf{yes}]_0$;
- (d) we replace r_{29} with the following:
 $r_{29} \equiv [\alpha_{3n+3m_k+1} d]_2 []_0 \rightarrow []_2 [\mathbf{no}]_0$;
- (e) cell-labeling rules: $r_{p;c}$ denotes r_p is used to label cell c ,
 $r_{30;1} \equiv [\beta_1]_3 [t'_1]_1 \rightarrow [v_1]_3 [v_1 t_1]_1$,
 $r_{31;1} \equiv [\beta'_1]_3 [f'_1]_1 \rightarrow [v_2]_3 [v_2 f_1]_1$.
 $r_{32;1} \equiv [\beta_i]_3 [t'_i v_l]_1 \rightarrow [v_{2l}]_3 [t_i v_{2l}]_1$, $l \in \{1, \dots, 2^n\}$, $2 \leq i \leq n$,
 $r_{33;1} \equiv [\beta'_i]_3 [f'_i v_l]_1 \rightarrow [v_{2l-1}]_3 [f_i v_{2l-1}]_1$, $l \in \{1, \dots, 2^n\}$, $2 \leq i \leq n$;
- (f) cleaning rules, unused objects (during computation) are dump to cell x
 $r_{34} \equiv [\mathbf{yes} v_l]_0 []_x \rightarrow []_0 [\mathbf{yes} v_l]_x$,
 $r_{35} \equiv []_3 [y]_1 \rightarrow [y']_3 []_1$,
 $r_{36} \equiv [y']_3 [] \rightarrow []_3 [y]_0$,
 $r_{37} \equiv [y]_0 []_x \rightarrow []_0 [y]_x$.

- R_Δ is the set of inter-component communication rules:

$$\begin{aligned}
R_{\text{bi-}\Delta^2} = & \{r'_1 \equiv (\langle 1, 0 \rangle, \mathbf{yes} v_l / \mathbf{yes} v_l, \langle 2, 0 \rangle), r'_2 \equiv (\langle 1, 0 \rangle, \mathbf{no} / \mathbf{no}, \langle 2, 0 \rangle), \\
& r'_3 \equiv (\langle 1, 0 \rangle, \lambda / \mathbf{no}, \langle 2, 0 \rangle), r'_4 \equiv (\langle 1, 0 \rangle, \mathbf{no} / \lambda, \langle 2, 0 \rangle), \\
& r'_5 \equiv (\langle 1, 0 \rangle, \mathbf{yes} / \lambda, 0), r'_6 \equiv (\langle 1, 0 \rangle, \mathbf{no} / \lambda, 0), r'_7 \equiv (\langle 2, 0 \rangle, \mathbf{yes} / \lambda, 0), \\
& r'_8 \equiv (\langle 2, 0 \rangle, \mathbf{no} / \lambda, 0)\}.
\end{aligned}$$

Note that each (uniform) recognizer tissue P system $\Pi_k(\langle n, m \rangle)$ in a dP scheme will process all Boolean formulas φ , which are in conjunctive normal form (CNF) with n variables and m clauses, where $\langle n, m \rangle = \frac{(n+m)(n+m+1)}{2} + n$, as long as appropriate input multiset $\text{cod}(\varphi)$ is supplied to each component system [39]. Furthermore, we will use non-deterministic maximal parallelism in the application of rules of the system. Thus, the correctness of the computations made by the component P systems of $2\text{-}\Delta_{TP_{ec}}$ is done [39].

In the construction of $2\text{-}\Delta_{TP_{ec}}$, labelling rules are introduced. Initially, both cells 1 and 3 contain a_1, a_2, \dots, a_n , which represent the variables in φ . After applying rules $r_{1,i;1}$ and $r_{1,i;3}$, we would have 2^n copies of cell 1 and 2^n copies of cell 3 in both Π_1 and Π_2 in $2\text{-}\Delta_{TP_{ec}}$. Each cell 1 contains a unique truth assignments of the n variables to be evaluated. Each cell 3 contains the corresponding sequence of β_i and β'_i , $1 \leq i \leq n$. Note that the number of t'_i and f'_i equals to the number of β_i and β'_i , respectively.

We need to show that our labelling of all cells 1 is unique to guarantee a consistent truth assignments by both component P systems in $2\text{-}\Delta_{TP_{ec}}$ for each variable in φ before the inter-component communication is done.

The set of labelling rules is composed of $r_{30;1}$ to $r_{33;1}$ of $2\text{-}\Delta_{TP_{ec}}$. The existence of β_i and β'_i in each of cell 3 is assured after applying $r_{1,i;3}$, $1 \leq i \leq n$. Also t'_i s and f'_i s are in each cell 1 after applying $r_{1,i;1}$.

Labelling rules can be expressed as follows. Given initial labels obtained by using $r_{30;1}$ and $r_{31;1}$. Let $r_{32;1}$ be the mapping $g_{t'}: l \mapsto 2l$, and $r_{33;1}$ be the mapping $g_{f'}: l \mapsto 2l - 1$. These mapping are bijections. Thus the unique labelling v_l of each cell 1 is obtained. Furthermore, each cell 1 labelled v_l contains distinct thruth assignments that makes true the formula φ .

The labelling procedure is done in $O(n)$ steps. Each component P system of $\Delta_{TP_{ec}}$ performs its evaluation individually in $3n + 3m_k + 2$ steps, where $m_k = \lceil \frac{m}{2} \rceil$. In particular, after $3n + 3m_k$ steps, E_{m_k+1} and v_l are found in each cell 1, which means the truth assignment for φ is satisfied. Rule r_{27} collects all pairs v_l and \mathbf{yes} in cell 3 at step $3n + 3m_k + 1$, then r_{28} releases pairs of v_l and \mathbf{yes} to the local environment of each component P system in the dP scheme. The communication rule r'_1 can be applied at the same time unused pairs of v_l and \mathbf{yes} during the communication will be dumped to cell x .

Finally, the object **yes** will be at the common environment after step $3n + 3m_k + 4$ or the object **no** will be at the common environment after step $3n + 3m_k + 3$. Note that at δ_h the object **yes** is in 0. \square

The succeeding results will measure the amount of communications in each component P system.

Theorem 15. *There exists a bi-directional P protocol $\Delta_{TP_{ec}}$ for solving the SAT problem under a balanced fixed-partition such that $ComN(\Delta_{TP_{ec}}) = 1$, $ComR(\Delta_{TP_{ec}}) = S$, $ComW(\Delta_{TP_{ec}}) = 4S$, where S is the number of satisfying truth assignment to the SAT problem.*

Proof: The dP scheme $2-\Delta_{TP_{ec}}$ from Theorem 14 decides φ using bi-directional P protocol. After the component P systems of $\Delta_{TP_{ec}}$ individually decide on their parts of the input, they would need to communicate their decisions to the other components for consistency of truth assignments. Since $2-\Delta_{TP_{ec}}$ is using an antiport-like inter-component communication rules, this requires only one communication. Each of the component P systems, if φ is satisfiable, then the system will certainly have some v_l . In particular, if $(\langle 1, 0 \rangle, v_l \mathbf{yes} / v_l \mathbf{yes}, \langle 2, 0 \rangle)$ is used by both component P systems Π_1 and Π_2 , which implies both of them obtained at least a satisfying truth assignment for φ .

Let T_k be the set of satisfying truth assignments obtained by Π_k , ($k = 1, 2$), then $T_1 \cap T_2$ is the set of satisfying truth assignments for φ . Let $|T_1 \cap T_2| = S$, then $ComN(\Delta_{TP_{ec}}) = 1$, $ComR(\Delta_{TP_{ec}}) = S$, $ComW(\Delta_{TP_{ec}}) = 4S$. \square

Theorem 16. *Let φ be any instance of SAT in CNF with m clauses and n variables. Then under a balanced fixed-partition, there is a two-way 2-P protocol $\Delta_{TP_{ec}}$ for solving the SAT problem such that $ComN(\Delta_{TP_{ec}}) = 2$, $ComR(\Delta_{TP_{ec}}) = S + T$, $ComW(\Delta_{TP_{ec}}) = 2(S + T)$, where S is the number of satisfying truth assignment to φ_1 , and T is the number of satisfying truth assignments of the SAT problem.*

Proof: The same $2-\Delta_{TP_{ec}}$ in Theorem 15 is used but rules in R_Δ will be as follows.

$$R_{2-\Delta^2} = \{r'_1 \equiv (\langle 1, 0 \rangle, v_l \mathbf{yes} / \lambda, \langle 2, 0 \rangle), r'_2 \equiv (\langle 1, 0 \rangle, \lambda / v'_l \mathbf{yes} \langle 2, 0 \rangle), \\ r'_3 \equiv (\langle 1, 0 \rangle, \mathbf{no} / \lambda, 0), r'_4 \equiv (\langle 2, 0 \rangle, y' \mathbf{yes} / \lambda, 0), r'_5 \equiv (\langle 2, 0 \rangle, \mathbf{no} / \lambda, 0)\}.$$

Furthermore, we add cell 4 to each component P system of $\Delta_{TP_{ec}}$. Then we have

$$\Delta_{TP_{ec}} = (\Gamma, 0, \Pi_1, \Pi_2, R_\Delta),$$

where $\Pi_k(\langle n, m \rangle) = (\Gamma_k, \Sigma_k, \mathcal{E}_k, \mathcal{M}_{k,1}, \mathcal{M}_{k,2}, \mathcal{M}_{k,3}, \mathcal{M}_{k,4}, \mathcal{M}_{k,x}, R_k, i_{k_{in}} = 2, i_{k_{out}} = 0)$, such that $\mathcal{M}_{k,4} = \{a_i \mid 1 \leq i \leq n\}$, and $\Gamma_k = \Gamma' \cup V \cup V' \cup \{\epsilon_i, \epsilon'_i, \kappa_i, \kappa'_i, \gamma_i, \gamma'_i\}$.

Similarly, we add the following rules in R_k :

Rule applied to cell 3 with objects $\beta_i, \beta'_i, \kappa_i$ and κ'_i
 $r_{1,i;3} \equiv [a_1]_3 \rightarrow [\beta_1 \kappa_1]_3 [\beta'_1 \kappa'_1]_3, \quad 1 \leq i \leq n.$

Rule applied to generate copies of cell 4 with objects γ'_i and ϵ'_i
 $r_{1,i;4} \equiv [a_i]_4 \rightarrow [\gamma'_i]_4 [\epsilon'_i]_4, \quad 1 \leq i \leq n.$

Cell-labeling rules for cell 4:

$$r_{30;4} \equiv [\kappa_1]_3 [\gamma'_1]_4 \rightarrow [v_1]_3 [v_1 \gamma_1]_4,$$

$$r_{31;4} \equiv [\kappa'_1]_3 [\epsilon'_1]_4 \rightarrow [v_2]_3 [v_2 \epsilon_1]_4,$$

$$r_{32;4} \equiv [\kappa_i]_3 [\gamma'_i v_l]_4 \rightarrow [v_{2l}]_3 [\gamma_i v_{2l}]_4, \quad l \in \{1, \dots, 2^n\}, \quad 2 \leq i \leq n,$$

$$r_{33;4} \equiv [\kappa'_i]_3 [\epsilon'_i v_l]_4 \rightarrow [v_{2l-1}]_3 [\epsilon_i v_{2l-1}]_4, \quad l \in \{1, \dots, 2^n\}, \quad 2 \leq i \leq n.$$

Additional rules

$$r_{38} \equiv [v_l^2 y]_0 [v_l]_4 \rightarrow [v_l]_0 [v_l' y']_4,$$

$$r_{39} \equiv []_0 [v_l' y']_4 \rightarrow [v_l' y']_0 []_4.$$

Let $\varphi = \varphi_1 \wedge \varphi_2$, where φ_1 is assigned to Π_1 and φ_2 is assigned to Π_2 . The inputs are placed in the appropriate cells of the component P systems of $\Delta_{TP_{ec}}$ in an encoded form. Solution must be made known to both component P systems of $\Delta_{TP_{ec}}$. Thus, the decision has been made known to both component P systems if object **yes** appeared in 0 or the common shared environment of Π_1 and Π_2 . Note that communications start from left to right, then from right to left.

When both components of $\Delta_{TP_{ec}}$ have already been produced all labels v_l in cell 1, and if object $E_{m_{k+1}}$ appears at least in a cell 1 (it means φ is satisfiable). $E_{m_{k+1}}$ together with v_l will be evolved to **yes**, v_l in $\langle k, 0 \rangle$, $k = 1, 2$. At this time, communication for the system commences.

Π_1 will communicate to Π_2 , all labels v_l of cell 1 that appear in its local environment use the rule $(\langle 1, 0 \rangle, v_l \text{ yes} / \lambda, \langle 2, 0 \rangle)$. Suppose there are S copies of v_l in $\langle 1, 0 \rangle$, hence $2S$ copies of objects have been communicated to Π_2 in one communication using S symport-like inter-component rules. Let T_1 and T_2 denote sets of satisfying assignments obtained by Π_1 and Π_2 , respectively, then $T_1 \cap T_2$ is the set of satisfying assignments for the SAT problem.

After performing rules r_{38} and r_{39} , Π_2 will send all v_l' and **yes** to Π_1 to inform the solution on the satisfiability of the SAT problem. At the same time, Π_2 sends objects **yes**, y' to 0. Finally, Π_1 is informed with the satisfying assignments for the SAT problem.

Therefore, $ComN(\Delta_{TP_{ec}}) = 2$, $ComR(\Delta_{TP_{ec}}) = S + T$, $ComW(\Delta_{TP_{ec}}) = 2(S + T)$. \square

Theorem 17. *Let φ be any instance of the SAT problem in CNF with m clauses and n variables. Then under a balanced fixed-partition, there is a one-way 2-P protocol $\Delta_{TP_{ec}}$ for the SAT problem such that $ComN(\Delta_{TP_{ec}}) = 1$, $ComR(\Delta_{TP_{ec}}) = S$, $ComW(\Delta_{TP_{ec}}) = 2S$, where S is the number of satisfying truth assignment of the SAT problem.*

Proof: The same 2- $\Delta_{TP_{ec}}$ in Theorem 16 is used.

Communications between Π_1 and Π_2 end after Π_1 sends its pairs v_l , **yes** to Π_2 . After using r_{39} , Π_2 sends copies of **yes** to 0 to declare satisfiability of the SAT problem.

At the end of computation/communications, both Π_1 and Π_2 know the labels of the satisfying truth assignments for the SAT problem, which requires only a single communication using S number of communication rules with a total of $2S$ objects, where S is the number of satisfying truth assignment of the SAT problem.

Therefore, $ComN(\Delta_{TP_{ec}}) = 1$, $ComR(\Delta_{TP_{ec}}) = S$, $ComW(\Delta_{TP_{ec}}) = 2S$. \square

Remark 18. Suppose we consider an unbalanced fixed-partition for the input of the SAT problem. Let $|P_1| = m_1$, and $|P_2| = m - m_1$ such that $|m_1 - m_2| \geq 2$. Then 2-P protocol $\Delta_{TP_{ec}}$ would need $3n + 3m' + 3$ steps to provide a decision for the SAT problem, where $m' = \max\{m_1, m_2\}$. Eventually, results on communication complexity (Theorems 15, 16 and 17) can be re-stated for the case of unbalanced fixed-partition.

5 Solving SAT by 3- $\Delta_{TP_{ec}}$

In this section, solution to the SAT problem will be presented using 3 components recognizer tissue P systems.

Theorem 19. *Let φ be any instance of the SAT problem in CNF with m clauses and n variables. Then under a balanced fixed-partition, there is a one-way 3-P protocol 3- $\Delta_{TP_{ec}}$ for SAT such that $ComN(\Delta_{TP_{ec}}) = 2$, $ComR(\Delta_{TP_{ec}}) = V'' + V'$, $ComW(\Delta_{TP_{ec}}) = 2(V'' + V')$, where V'' is the number of satisfying truth assignments to φ_2 , and V' is the number of satisfying truth assignments of Π_1 for φ_1 .*

Proof: Let $3\text{-}\Delta_{TPec} = (\Gamma, 0, \Pi_1, \Pi_2, \Pi_3, R_\Delta)$ be a dP scheme, where each Π_k ($k = 1, 2, 3$) is the same as those in Theorem 16. Each Π_k has almost the same set of rules presented in Theorem 16 in processing input instance φ of SAT. In this model, the following rule is added:

$$r_{40} \equiv [v'_l v_l y]_0 [v_l]_4 \rightarrow [v_l]_0 [v'_l y']_4.$$

Consequently, the following inter-component communication rules for $3\text{-}\Delta_{TPec}$ will be used.

$$\begin{aligned} R_{1-\Delta^3} = & \{r'_1 \equiv (\langle 1, 0 \rangle, v_l \text{ yes } / \lambda, \langle 2, 0 \rangle), r'_2 \equiv (\langle 2, 0 \rangle, v'_l, \text{ yes } / \lambda \langle 3, 0 \rangle), \\ & r'_3 \equiv (\langle 3, 0 \rangle, y' \text{ yes } / \lambda, 0), r'_4 \equiv (\langle 3, 0 \rangle, \text{ no } / \lambda, 0), \\ & r'_5 \equiv (\langle 2, 0 \rangle, \text{ no } / \lambda, 0), r'_6 \equiv (\langle 1, 0 \rangle, \text{ no } / \lambda, 0)\}. \end{aligned}$$

Communication between components of $3\text{-}\Delta_{TPec}$ is done successively from Π_1 to Π_2 , then from Π_2 to Π_3 . After each component processed their part of the input, Π_1 starts communication with Π_2 by sending all labels of cell 1. Π_2 obtained all these v_l , which are labels of cell 1 that provide satisfying truth assignments for φ from Π_1 . Let T_1 be the set of all labels v_l of cell 1, if $|T_1| = V'$, then Π_1 sent $2V'$ copies of object to Π_2 in one step.

Now Π_3 obtained from Π_2 copies of object v'_l and **yes** after $3n + 3m_k + 6$ steps. Each v'_l is a label of a satisfying truth assignment made by Π_1 and Π_2 , hence all copies of objects $v'_l, v_l, v_l', \text{ yes}$ are contained in $\langle 3, 0 \rangle$ after Π_2 sent objects $v'_l, \text{ yes}$ to Π_3 . Π_3 uses r_{40} and r_{39} to prepare using $(\langle 3, 0 \rangle, y' \text{ yes } / \lambda, 0)$ to declare satisfiability of φ . The number of y' is equal to the number of satisfying truth assignments of φ . In particular, the number of y' is equal to $|T_1 \cap T_2 \cap T_3|$, where T_1, T_2 , and T_3 are sets of satisfying truth assignments evaluated by Π_1, Π_2 , and Π_3 , respectively.

Let $|T_1| = V'$, and $|T_2| = V''$. Finally, we have $ComN(3\text{-}\Delta_{TPec}) = 2$, $ComR(3\text{-}\Delta_{TPec}) = V'' + V'$, and $ComW(3\text{-}\Delta_{TPec}) = 2V'' + 2V'$. \square

Theorem 20. *Let φ be any instance of the SAT problem in CNF with m clauses and n variables. Then under a balanced fixed-partition, there is a two-way 3-P protocol $3\text{-}\Delta_{TPec}$ for the SAT problem such that $ComN(3\text{-}\Delta_{TPec}) = 4$, $ComR(3\text{-}\Delta_{TPec}) = 2S + V' + V''$, $ComW(\Delta_{3\text{-}TPec}) = 2(S + V' + V'')$, where S is the number of satisfying truth assignment to φ , V' is the number of satisfying assignments of Π_1 for φ_1 , and $V'' = |T_1 \cap T_2|$, where T_1 and T_2 are satisfying truth assignments of φ_1 and φ_2 , respectively.*

Proof: The $3\text{-}\Delta_{TPec}$ in Theorem 19 is used but the set of inter-component communication rules $R_\Delta = R_{2-\Delta^3}$ uses only the symport-like communication rules. Specifically,

$$R_{2-\Delta^3} = R_{1-\Delta^3} \cup \{(\langle 2, 0 \rangle, \lambda / v'_l, \langle 3, 0 \rangle), (\langle 1, 0 \rangle, \lambda / v'_l, \langle 2, 0 \rangle)\}.$$

From Π_1 , it is easy to know that $2V'$ copies of object are sent to Π_2 , where $V' = |T_1|$, T_1 being the set of satisfying truth assignments for φ_1 evaluated by Π_1 .

Using r_{40} and r_{39} , Π_2 will eventually send objects v'_l and **yes** to Π_3 . The total amount of objects is equal to $2V''$ in a single communication. Π_3 realizes S labels that give satisfying truth assignments for φ , where S is the total number of labels that are common to all component P systems.

After $3n + 3m_k + 10$ steps, y' and **yes** will be sent by Π_3 to 0, simultaneously, objects v'_l and **yes** are sent to Π_2 . Hence Π_2 sends the same copies of objects to Π_1 . The communication going back from Π_3 to Π_1 requires $2S$ copies of objects using $2S$ rules in two communications.

Therefore, $ComN(3\text{-}\Delta_{TPec}) = 4$, $ComR(3\text{-}\Delta_{TPec}) = S + V' + V''$, and $ComW(3\text{-}\Delta_{TPec}) = S + 2V' + 2V''$. \square

Theorem 21. *Let φ be any instance of the SAT problem in CNF with m clauses and n variables. Then under a balanced fixed-partition, there is a bi-directional 3-P protocol $3\text{-}\Delta_{TPec}$ for the SAT problem such that $ComN(3\text{-}\Delta_{TPec}) = 1$, $ComR(3\text{-}\Delta_{TPec}) = S$, $ComW(\Delta_{3\text{-}TPec}) = 6S$, where S is the number of satisfying truth assignment to φ .*

Proof: The 3-P protocol $3\text{-}\Delta_{TPec}$ used in this proof will have component P systems similar to that in Theorem 15, but we use the following additional rules:

$$r_{41} \equiv [v_l \text{ yes}]_3 [v_l]_4 \rightarrow []_3 [v_l^3 \text{ yes}^3]_4, \text{ and } r_{42} \equiv []_0 [v_l' \text{ yes}]_4 \rightarrow [v_l \text{ yes}]_0 []_4,$$

and the set of inter-component communication rules R_Δ as follows:

$$\begin{aligned} R_{\text{bi-}\Delta^3} = & \{r'_1 \equiv (\langle 1, 0 \rangle, \text{yes } v_l / \text{yes } v_l, \langle 2, 0 \rangle), r'_2 \equiv (\langle 1, 0 \rangle, \text{yes } v_l / \text{yes } v_l, \langle 3, 0 \rangle), \\ & r'_3 \equiv (\langle 2, 0 \rangle, \text{yes } v_l / \text{yes } v_l, \langle 3, 0 \rangle), \\ & r'_4 \equiv (\langle 1, 0 \rangle, y \text{ yes} / \lambda, 0), r'_5 \equiv (\langle 2, 0 \rangle, y \text{ yes} / \lambda, 0), r'_6 \equiv (\langle 3, 0 \rangle, y \text{ yes} / \lambda, 0), \\ & r'_7 \equiv (\langle 1, 0 \rangle, \text{no} / \lambda, 0), r'_8 \equiv (\langle 2, 0 \rangle, \text{no} / \lambda, 0), r'_9 \equiv (\langle 3, 0 \rangle, \text{no} / \lambda, 0)\}. \end{aligned}$$

Furthermore, each component of $3\text{-}\Delta_{TPec}$ is as follows:

$$\Pi_k(\langle n, m \rangle) = (\Gamma_k, \Sigma_k, \mathcal{E}_k, \mathcal{M}_{k,1}, \mathcal{M}_{k,2}, \mathcal{M}_{k,3}, \mathcal{M}_{k,4}, \mathcal{M}_{k,x}, R_k, i_{k_{in}} = 2, i_{k_{out}} = 0),$$

such that $\mathcal{M}_{k,4} = \{a_i \mid 1 \leq i \leq n\}$ and $\Gamma_k = \Gamma' \cup V \cup V' \cup \{\epsilon_i, \epsilon'_i, \kappa_i, \kappa'_i, \gamma_i, \gamma'_i\}, r_{34} \notin R_k$.

In this modified $3\text{-}\Delta_{TPec}$, the additional rules allow each component P system to triple its v_l and **yes** in order to prepare for a simultaneous antiport-like communications. If φ is satisfiable, then rules r'_1, r'_2 , and r'_3 could be used. Simultaneously, by all component P systems send y , **yes** to 0. Since the communication is bi-directional, this is done in one step, using S rules and total of 6 objects. Note that S is the number of satisfying assignments for φ .

Therefore, $\text{ComN}(3\text{-}\Delta_{TPec}) = 1, \text{ComR}(3\text{-}\Delta_{TPec}) = S, \text{ComW}(\Delta_{3\text{-}TPec}) = 6S$, where S is the number of satisfying truth assignment to φ . \square

6 Relative performance of $k\text{-}\Delta_{TPec}$

The relative performance and parallelizability of $k\text{-}\Delta_{TPec}$ is considered in this section. The concept of *weak parallelizability* introduced in [21] is also considered.

A problem L is said to be (k, m) -weakly ComX parallelizable, for some $k \geq 2, m \geq 1$ and $X \in \{N, R, W\}$, if there is a dP scheme Δ with k components and there is a finite $F_\Delta \subseteq L$ such that each string $x \in L - F_\Delta$ can be written as $x = x_1 x_2 \cdots x_k$, such that $||x_i| - |x_j|| \leq 1$ for all $1 \leq i, j \leq k$, each component Π_i takes as input the string $x_i, 1 \geq i \leq k$ and string x is accepted by Δ in a halting computation δ such that $\text{ComX} \leq m$. A problem L is called *weakly ComX parallelizable* if it is (k, m) -weakly ComX parallelizable for some $k \geq 2, m \geq 1$.

In the case of $k\text{-}\Delta_{TPec}, k = 2, 3$ deciding on the SAT problem, the following results on parallelizability are obtained. In particular, results presented in Section 3 implies the following.

Theorem 22. *Let $\text{SAT} = \{\varphi \mid \varphi \text{ has } n \text{ variables and } m \text{ clauses}\}$.*

1. *Let $2\text{-}\Delta_{TPec}$ be a uniform bi-directional 2-P protocol for SAT under balanced fixed-partition, then SAT is $(2, r)$ -weakly ComX parallelizable, where $(r, \text{ComX}) \in \{(1, \text{ComN}), (S, \text{ComR}), (4S, \text{ComW})\}$, S is the number of satisfying truth assignments for φ .*
2. *Let $2\text{-}\Delta_{TPec}$ be a uniform two-way 2-P protocol for SAT under balanced fixed-partition, then SAT is $(2, r)$ -weakly ComX parallelizable, where $(r, \text{ComX}) \in \{(2, \text{ComN}), (S + T, \text{ComR}), (2(S + T), \text{ComW})\}$, where S is the number of satisfying truth assignment to φ_1 , and T is the number of satisfying truth assignments for φ .*
3. *Let $2\text{-}\Delta_{TPec}$ be a uniform one-way 2-P protocol for SAT under balanced fixed-partition, then SAT is $(2, r)$ -weakly ComX parallelizable, where $(r, \text{ComX}) \in \{(1, \text{ComN}), (S, \text{ComR}), (2S, \text{ComW})\}$, where S is the number of satisfying truth assignment of φ_1 .*

In the case of $3\text{-}\Delta_{TPec}$ for solving the SAT problem under balanced fixed-partition, the results in Section 5 implies the following.

Theorem 23. *Let $SAT = \{\varphi \mid \varphi \text{ has } n \text{ variables and } m \text{ clauses}\}$.*

1. *Let $2\text{-}\Delta_{TPec}$ be a uniform bi-directional 3-P protocol for SAT under balanced fixed-partition, then SAT is $(3, r)$ -weakly ComX parallelizable, where $(r, ComX) \in \{(1, ComN), (S, ComR), (6S, ComW)\}$, where S is the number of satisfying truth assignments for φ .*
2. *Let $2\text{-}\Delta_{TPec}$ be a uniform two-way 3-P protocol for SAT under balanced fixed-partition, then SAT is $(3, r)$ -weakly ComX parallelizable, where $(r, ComX) \in \{(4, ComN), (2S + V' + V'', ComR), (2(S + V' + V''), ComW)\}$, where S is the number of satisfying truth assignment to φ , V' is the number of satisfying assignments of Π_1 for φ_1 , and $V'' = |T_1 \cap T_2|$, where T_1 , and T_2 are satisfying truth assignments of φ_1 , and φ_2 , respectively.*
3. *Let $2\text{-}\Delta_{TPec}$ be a uniform one-way 3-P protocol for SAT under balanced fixed-partition, then SAT is $(3, r)$ -weakly ComX parallelizable, where $(r, ComX) \in \{(2, ComN), (V'' + V', ComR), (2(V'' + V'), ComW)\}$, where V'' is the number of satisfying truth assignment to φ_2 , and V' is the number of satisfying assignments of Π_1 for φ_1 .*

The relative efficiency of performance of $k\text{-}\Delta_{TPec}$ ($k = 2, 3$) can also be viewed with respect to its computation time spent solving a problem. In this respect, $k\text{-}\Delta_{TPec}$ will be compared to the efficient solution presented in [39]. Let $TIME_{\Pi(\langle n, m \rangle)}(n, m)$ be the running time of $\Pi(\langle n, m \rangle)$, and $TIME_{\Delta_{TPec}}(n, m)$ denotes the running time of Δ_{TPec} .

In [39], $TIME_{\Pi(\langle n, m \rangle)}(n, m) = 2n + 3m + 2$, while Theorem 14 gives $TIME_{2\text{-}\Delta_{TPec}}(n, m) = 3n + 3\frac{m}{k} + 4$. The following limit represents the speed up ratio between $\Pi(\langle n, m \rangle)$ and $2\text{-}\Delta_{TPec}(n, m)$.

$$\lim_{n \rightarrow \infty} \frac{TIME_{\Pi(\langle n, m \rangle)}(n, m)}{TIME_{2\text{-}\Delta_{TPec}}(n, m)} = \lim_{n \rightarrow \infty} \frac{2n + 3m + 2}{3n + 3\frac{m}{k} + 4}.$$

The value of this limit is required to be at least 2, to imply improvements of the computation by $2\text{-}\Delta_{TPec}(n, m)$ compared with that by $\Pi(\langle n, m \rangle)(n, m)$ solving the same problem.

Let $k = 2$, and $m = n$, the speed-up ratio is:

$$\lim_{n \rightarrow \infty} \frac{TIME_{\Pi(\langle n, m \rangle)}(n, m)}{TIME_{2\text{-}\Delta_{TPec}}(n, m)} = \lim_{n \rightarrow \infty} \frac{5n + 2}{4.5n + 4} \approx 1.11.$$

This suggests that for any $k \geq 2$, as long as $m \leq n$, $k\text{-}\Delta_{TPec}$ could not do significant advantage compared with $\Pi(\langle n, m \rangle)$ for solving SAT. It can also be observed that if for any k , $n = m^2$, we would have

$$\lim_{n \rightarrow \infty} \frac{TIME_{\Pi(\langle n, m \rangle)}(n, m)}{TIME_{\Delta_{TPec}}(n, m)} = \lim_{m \rightarrow \infty} \frac{2m^2 + 3m + 2}{3m^2 + 3\frac{m}{k} + 4} < \frac{2}{3}.$$

This would mean no parallelism.

If we let $m = n^2$, then speed-up ratio becomes

$$\lim_{n \rightarrow \infty} \frac{TIME_{\Pi(\langle n, m \rangle)}(n, m)}{TIME_{\Delta_{TPec}}(n, m)} = \lim_{n \rightarrow \infty} \frac{2n + 3n^2 + 2}{3n + \frac{3}{k}n^2 + 4} = k \geq 2.$$

This shows that $k\text{-}\Delta_{TP_{ec}}$ computes in at least half the required number of steps by $\Pi(\langle n, m \rangle)$, if $m \geq n^2$, for any k .

The uniform recognizer tissue P systems in [39] may not be the optimal uniform recognizer tissue P systems for solving the SAT problem, that is, deciding SAT with the smallest possible steps, but it is efficient enough to compare the relative performance of $k\text{-}\Delta_{TP_{ec}}$ for solving SAT.

7 Conclusions and discussions

In this work, a distributed P scheme that solves instances φ of SAT is presented. The power of the recognizer tissue P systems with evolutionary communication rules and cell division from [39] is capitalized in a dP scheme. Labelling of all cells 1 after cell division is suggested to give precise and exact decision on the satisfiability of φ . Moreover, $\Delta_{TP_{ec}}$ requires that whatever is the decision for φ , all component P systems know the decision. Two possible types of communication that $\Delta_{TP_{ec}}$ could be performed, namely, *antiport-like inter-component communication* and *symport-like inter-component communication*. Thus, the concept of a P protocol is introduced. Taking into account the types of inter-component communications on dP scheme, one-way P protocol, two-way P protocol and bi-directional P protocol are defined. The concept of a uniform P protocol is also mentioned. The idea of balanced and unbalanced partitions are also presented and, in particular, a so-called (un)balanced fixed-partition is considered in distributing parts of the input to component P systems of dP scheme.

It is shown that under a balanced fixed-partition $k\text{-}\Delta_{TP_{ec}}$, could be able to decide on the satisfiability of any instance φ of SAT using only one communication under a bi-directional $k\text{-}P$ protocol. The number of inter-component rules is the number of satisfying truth assignments for φ . But the number of objects sent by the k component P systems increases with respect to k . In the case $k = 2, 3$, we obtained $ComN(2\text{-}\Delta_{TP_{ec}}) = 1$, $ComR(2\text{-}\Delta_{TP_{ec}}) = S$, $ComW(2\text{-}\Delta_{TP_{ec}}) = 4S$, and $ComN(3\text{-}\Delta_{TP_{ec}}) = 1$, $ComR(3\text{-}\Delta_{TP_{ec}}) = S$, $ComW(\Delta_{3\text{-}TP_{ec}}) = 6S$, where S is the number of satisfying truth assignment to φ .

Notice that $k\text{-}\Delta_{TP_{ec}}$ is a uniform dP scheme, that is, each component P system Π_k has (almost) the same set of rules being implemented during every computation. It is also assume that each cell in tissue P systems is connected to every other cells in the system and can communicate directly with each other. The only trade-off is extra steps for each component to reproduce the objects to be communicated using bi-directional mode of communication. This is polynomial with respect to the number of component P systems. This implies that under $k\text{-}\Delta_{TP_{ec}}$, SAT is $(k, 1)$ -weakly *ComN parallelizable*, for all k . Note that this invariance with respect to *weakly ComN* is obtained under a balanced fixed-partition of input, fixed encoding and with a bi-directional $k\text{-}P$ protocol.

The same invariance could be observed in the case of (k, S) -weakly *ComR parallelizability* of $k\text{-}\Delta_{TP_{ec}}$ under a balanced fixed k -partition using bi-directional communication mode, for any k . Notice that the k components P systems in $k\text{-}\Delta_{TP_{ec}}$ will have to produce k copies of the labels of cell with satisfying truth assignments of their respective part of the input. Eventually, $k\text{-}\Delta_{TP_{ec}}$ uses the antiport-like inter component communication rule that matches these labels together with **yes**. Finally every cell in $k\text{-}\Delta_{TP_{ec}}$ sends object **yes** to 0 to signal the end of the computation and decided the satisfiability of φ .

Note that in Remark 18, it was stated that Theorem 15 and Theorem 16 could be re-stated in the case of unbalanced partition. Then at least for $k = 2, 3$, SAT is $(k, 1)$ -weakly *ComN parallelizable* and (k, S) -weakly *ComR parallelizable* under an unbalanced fixed-partition. It is believe that SAT is $(k, 1)$ -weakly *ComN parallelizable* and is also (k, S) -weakly *ComR parallelizable* for any k under an unbalanced k fixed-partition.

Statement 1 of both Theorem 22 and Theorem 23 shows that SAT belongs to the class of problems that could be solved by uniform k - $\Delta_{TP_{ec}}$, $k = 2, 3$ with $ComW(2-\Delta_{TP_{ec}}) = 4S$, and $ComW(3-\Delta_{TP_{ec}}) = 6S$, where S is the number of satisfying truth assignment to φ , which implies that the objects needed to be communicated by the system increases with the number components. Note that a uniform $3-\Delta_{TP_{ec}}$ for SAT needs more $2S$ objects to decide the satisfiability of φ compared with $2-\Delta_{TP_{ec}}$. Using the uniform $k-\Delta_{TP_{ec}}$ in this paper, it might be reasonable to believe that SAT may be (k, s) -weakly $ComW$ parallelizable, but it is not $(k + 1, s)$ -weakly $ComR$ parallelizable, for any k and for some s .

In the case of one-way and two-way uniform k -P protocols under balanced fixed-partition ($k = 2, 3$), it was demonstrated that the total amount of objects to be communicated and the total number of inter-component rules are increased with respect to the number of component P systems of $k-\Delta_{TP_{ec}}$. These results suggest that $ComX$, $X \in \{N, R, W\}$ is directly proportional to k . In particular, SAT belongs to the class of problems that is $(2, r)$ -weakly $ComX$ parallelizable, which do not belong to the class of problems that are $(3, r)$ -weakly $ComX$ parallelizable, where $(r, ComX) \in \{(r, ComN), (r, ComR), (r, ComW)\}$. It is of interest to know if these observed relations between $2-\Delta$ and $3-\Delta$ could be extended to $k-\Delta$ and $(k + 1)-\Delta$ with one-way and two-way uniform k P protocols under balanced fixed-partition.

It is also realized that the amount of clauses related to the number of variables is quite necessary in order to obtain efficiency in using $k-\Delta_{TP_{ec}}$ to solve SAT. In particular, if $m \leq n$, the relative efficiency of $k-\Delta_{TP_{ec}}$ cannot be equal to 2. This is regardless if we increase the number k of component P systems. But at $m = n^2$, we obtain a reasonable relative efficiency k , for any $k \in O(n)$. Notice here that this efficiency is an upper bound of the precise efficiency we wanted to obtain, since $k-\Delta_{TP_{ec}}$ is compared only to a particular $\Pi(\langle n, m \rangle)$ for solving SAT.

In [21], a problem L is said to be (k, r, s) -efficiently $ComX$ parallelizable, for some $k \geq 2, r \geq 1, s \geq 2$, and $X \in \{N, R, W\}$, if it is (k, r) -weakly $ComX$ parallelizable, and there is a dP scheme Δ such that

$$\lim_{n \rightarrow \infty} \frac{TIME_{\Pi}(x)}{TIME_{\Delta}(x)} \geq s,$$

for all P systems Π such that $L = L(\Pi)$. Moreover, $TIME_{\Pi}(x)$ is the smallest number of steps need for Π to accept string x should be estimated with respect to all Π for L , while $TIME_{\Delta}(x)$ is just given by means of a construction of a suitable dP scheme Δ . It might be reasonable to believe that SAT is (k, r, s) efficiently $ComX$ parallelizable, where $(r, ComX) \in \{(r_1, ComN), (r_2, ComR), (r_3, ComW)\}$, for some real numbers $r_i, i = 1, 2, 3; s \leq k$ is the speed up ratio and k is the number of components in the uniform dP scheme under bi-directional, one-way and two-way uniform k -P protocol.

Notice that in order to minimize the amount of objects to be communicated proper, encoding of objects is necessary. We need not to communicate the whole multiset of objects, but an encoded version of them. This encoding add-up to the time and number of cells to be used by component P systems in the systems. In the case of this paper, cell labelling is proposed to encode the truth assignment uniquely to maintain consistency of assigning truth values to variable being evaluated by the whole systems. In order to keep the use of rules efficiently, we have to expect to produce at most exponential amount of cells. Finally, we suggest that one of possible path to take in this line of research is to minimize the amount of objects to be communicated by component P systems in solving problems, keeping the performance of component P systems within reasonable efficiency.

Uniform P protocols under balanced fixed-partition are the ones considered, and remarked on the unbalanced fixed-partition for solving SAT is provided. It would be nice to consider what may be called *optimal-partition*, where we design partition of the objects of the input and see how it fared with fixed-partition with respect to communication measures. Non-uniform k -P

protocol solving hard problems might also be a nice direction to pursue. By non-uniform, means allowing each component P system to perform what it thinks necessary with respect to the input part. Furthermore, it is of interest to consider communication resources with respect to some communication P protocols or dP schemes for solving other hard problems.

Acknowledgements

The work was supported by National Natural Science Foundation of China (61320106005, 61602192, and 61772214), China Postdoctoral Science Foundation (2016M600592 and 2017T100554), and the Innovating Scientists and Technicians Troop Construction Projects of Henan Province (154200510012). H. Adorna was also supported by Semirara Mining Corporation, Inc. Professorial Chair of the College of Engineering, U.P. Diliman, DOST-Engineering Research and Development for Technology Program grant, and an OVCRD RLC grant 2017-2018.

Bibliography

- [1] Adorna, H., Păun, Gh. Pérez-Jiménez, M.J. (2010): On Communication Complexity in Evolution-Communication P Systems, *Rom. J. Inf. Sci. Tech.*, 13(2), 113–130, 2010.
- [2] Alhazov, A. (2004): On Determinism of Evolution-Communication P Systems, *J. Univers. Comput. Sci.*, 10(5), 502–508, 2004.
- [3] Buño, K., Adorna, H., Pan, L. Song, B.(2017): Communicaton Complexity of Distributed Tissue-Like P Systems for Solving SAT Problem, *Proc. of the 18th International Conference on Membrane Computing*, Bradford, UK, 373–383, 2017.
- [4] Buño, K. Cabarle, F., Adorna, H., Calabia, M.(2018): Solving N-Queens Problem using dP Systems with Active Membranes, *Theor. Comput. Sci.*, 2018.
- [5] Cavaliere, M. (2003): Evolution-Communication P Systems, In: *Păun, Gh. et al, (Eds.) LNCS*, Springer, Heidelberg, 2597, 134–145, 2003.
- [6] Csuhaj-Varjú, E., Margenstern, M., Vazil, G., Verlan, S. (2007): On Small Universal Antipport P System, *Theor. Comput. Sci.*, 372(2-3), 152–164, 2007.
- [7] Csuhaj-Varjú, E., Vaszil, G. (2012): Finite dP Automata Versus Multi-head Finite Automata. In: *Gheorghe, M. et al. (Eds.): LNCS*, Springer, Heidelberg, 7184, 120–138, 2012.
- [8] Díaz-Pernil, D., Peña-Cantillana, F., Gutiérrez-Naranjo, M.A. (2013): A Parallel Algorithm for Skeletonizing Images by Using Spiking Neural P Systems, *Neurocomputing*, 115, 81–91, 2013.
- [9] Dzitac, I. (2015): Impact of Membrane Computing and P Systems in ISI WoS. Celebrating the 65th Birthday of Gheorghe Păun, *Int. J. Comput. Commun*, 10(5), 617–626, 2015.
- [10] Elias, S., Gokul, V, Krithivasan, K., Gheorghe, M., Zhang, G. (2012): A Variant of Distributed P Systems for Real Time Cross Layer Optimization, *J. Univers. Comput. Sci.*, 18(13), 1760–1781, 2012.
- [11] Frisco, P., Govan, G., Leporati, A. (2012): Asynchronous P Systems with Active Membranes, *Theor. Comput. Sci.*, 429, 74–86, 2012.

-
- [12] Gazdag, Z. (2014): Solving SAT by P Systems with Active Membranes in Linear Time in the Number of Variables. In: *Alhazov, A. et al. (Eds.) LNCS*, Springer, Heidelberg, 8340, 189–205, 2014.
- [13] Gutiérrez-Naranjo, M.A., Martínez-del-Amor, M.A., Pérez-Jurtado, I., Pérez-Jiménez, M.J. (2009): Solving N-Queens Puzzle with P Systems. In: *Proc. of the 7th Brainstorming Week on Membrane Computing*, Sevilla, Spain, 199–210, 2009.
- [14] Hernandez, N., Juayong, R., Adorna, H. (2014): On the Communication Complexity of Some Hard Problems in ECPe Systems. In: *Alhazov, A. et al. (Eds.): LNCS*, Springer, Heidelberg, 8340, 206–224, 2014.
- [15] Macías-Ramos, L.F., Valencia-Cabrera, L., Song, B., Pan, L., Pérez-Jiménez, M.J. (2015): A P-Lingua Based Simulator for P Systems with Symport/Antiport Rules, *Fund. Informa.*, 139(2), 211–277, 2015.
- [16] Martínez-del-Amor, M.A., García-Quismondo, M., Macías-Ramos, L.F., Valencia-Cabrera, L., Riscos-Núñez, A., Pérez-Jiménez, M.J. (2015): Simulating P Systems on GPU Devices: A Survey, *Fund. Informa.*, 136(3), 269–284, 2015.
- [17] Pan, L., Păun, Gh. (2009): Spiking Neural P Systems with Anti-Spikes, *Int. J. Comput. Commun.*, 4(3), 273–282, 2009.
- [18] Pan, L., Pérez-Jiménez, M.J. (2010): Computational Complexity of Tissue-Like P Systems, *J. Complexity*, 26(3), 296–315, 2010.
- [19] Păun, Gh. (2000): Computing with Membranes, *J. Comput. Syst. Sci.*, 61(1), 108–143, 2000.
- [20] Păun, Gh. (2016): Membrane Computing and Economics: A General View, *Int. J. Comput. Commun.*, 11(1), 105–112, 2016.
- [21] Păun, Gh., Pérez-Jiménez, M.J. (2010): Solving Problem in a Distributed Way in Membrane Computing: dP Sytems, *Int. J. Comput. Commun.*, 5, 238–259, 2010.
- [22] Păun, Gh., Pérez-Jiménez, M.J. (2012): An Infinite Hierarchy of Languages Defined by dP Systems, *Theor. Comput. Sci.*, 431, 4–12, 2012.
- [23] Păun, Gh. Rozenberg, G., Salomaa, A. (Eds.)(2010): *The Oxford Handbook of Membrane Computing*, Oxford University Press, New York, 2010.
- [24] Peng, H., Wang, J., Pérez-Jiménez, M.J., Riscos-Núñez, A. (2015): An Unsupervised Learning Algorithm for Membrane Computing, *Inf. Sci.*, 304, 80–91, 2015.
- [25] Song, T., Pan, L. (2016): Spiking Neural P Systems with Request Rules, *Neurocomputing*, 193, 193–200, 2016.
- [26] Song, T., Pan, L., Păun, Gh. (2013): Asynchronous Spiking Neural P Systems with Local Synchronization, *Inf. Sci.* 219, 197–207, 2013.
- [27] Song, B., Pan, L. (2016): The Computational Power of Tissue-Like P Systems with Promoters, *Theor. Comput. Sci.*, 641, 43–52, 2016.
- [28] Song, B., Song, T., Pan, L. (2017): A Time-Free Uniform Solution to Subset Sum Problem by Tissue P Systems with Cell Division, *Math. Struct. Comp. Sci.*, 27(1), 17–32, 2017.

- [29] Song, B., Zhang, C., Pan, L. (2017): Tissue-Like P Systems with Evolutional Symport/Antiport Rules, *Inf. Sci.*, 378, 177–193, 2017.
- [30] Valencia-Cabrera, L., Orellana-Martín, D., Martínez-del-Amor, M.A., Riscos-Núñez, A., Pérez-Jiménez, M.J. (2017): Cooperation in Transport of Chemical Substances: A Complexity Approach within Membrane Computing, *Fund. Informa.*, 154(1-4), 373–385, 2017.
- [31] Valencia-Cabrera, L., Orellana-Martín, D., Martínez-del-Amor, M.A., Riscos-Núñez, A., Pérez-Jiménez, M.J. (2017): Reaching Efficiency through Collaboration in Membrane Systems: Dissolution, Polarization and Cooperation, *Theor. Comput. Sci.*, 701, 226–234, 2017.
- [32] Valencia-Cabrera, L., Wu, T., Zhang, Z., Pan, L., Pérez-Jiménez, M.J. (2016): A Simulation Software Tool for Cell-Like Spiking Neural P Systems, *Rom. J. Inf. Sci. Tech.*, 20(1), 71–84, 2016.
- [33] Wu, T., Zhang, Z., Păun, Gh., Pan, L. (2016): Cell-Like Spiking Neural P Systems, *Theor. Comput. Sci.*, 623, 180–189, 2016.
- [34] Yao, A.C. (1979): Some Complexity Questions Related to Distributed Computing, In: *Proceedings of the eleventh annual ACM symposium on Theory of computing*, 209–213, 1979.
- [35] Zandron, C., Leporati, A., Ferretti, C., Mauri, G., Pérez-Jiménez, M.J. (2008): On the Computational Efficiency of Polarizationless Recognizer P Systems with Strong Division and Dissolution, *Fund. Informa.*, 87(1), 79–91, 2008.
- [36] Zhang, G., Rong, H., Neri, F., Pérez-Jiménez, M.J. (2014): An Optimization Spiking Neural P System for Approximately Solving Combinatorial Optimization Problems, *Int. J. Neural Syst.*, 24, 1–16, 2014.
- [37] Zhang, X., Pan, L., Păun, A. (2015): On the Universality of Axon P Systems, *IEEE T. Neur. Net. Lea.*, 26(11), 2816–2829, 2015.

Appendix: Some Tables for Inter-component Communications

The following tables below show how communication between component P systems transfer. The table of communications starts when the systems already obtained a satisfying truth assignments of their respective input parts. Notice that Table 6 which is continued in Table 7 starts at step 4. This initially started when E_{m_k+1} appeared in any of the cell 1, which will be the step 1 of the table.

Below are tables for cases where there are at most three component recognizer tissue P systems with evolutionary communication rules and cell division. one-way, two-way and bi-directional P protocol are considered below. Notice that the set of communication rules differs per kind of P protocol model. The set of rules used by each component P systems are mostly based on the results in [39]. Variations on rules R_k for each k depends on the P communication mode required of the systems.

The tables provide labels of each column and each row provides information on specific action transferred with respect to the preceding row.

One would notice that the set of inter-component rules of other P communication mode are similar as stated in the proof of some of the Theorems.

Table 1: Expanded Communication Configuration for k - Δ . Rejecting Communication (1).

step	rule	cell 1	cell 2	cell 3	cell x	$\langle 1, 0 \rangle$	0	$\langle 2, 0 \rangle$	cell x	cell 3	cell 2	cell 1
1	r_{26}	v_l	α_p, d	v_l						v_l	α_p	v_l
2	r_{29}	y	α_{p+1}, d	$v_l,$						v_l, yes	α_{p+1}, d	y
3	r'_6					no		v_l, yes		y'	α_{p+2}, d	
4							no	v_l, yes		y'	α_{p+3}, d	

 Table 2: Expanded Communication Configuration for k - Δ . Rejecting Communication (2).

step	rule	cell 1	cell 2	cell 3	cell x	$\langle 1, 0 \rangle$	0	$\langle 2, 0 \rangle$	cell x	cell 3	cell 2	cell 1
1	r_{26}	v_l	α_p, d	v_l						v_l	α_p	v_l
2	r_{29}	y	α_{p+1}, d	$v_l,$						v_l	α_{p+1}, d	y
3	r'_2	y				no		no				y
4		y					no					y

 Table 3: Expanded Communication Configuration for $\text{bi-}\Delta^2$. Accepting Communication.

step	rule	cell 1	cell 2	cell 3	cell x	$\langle 1, 0 \rangle$	0	$\langle 2, 0 \rangle$	cell x	cell 3	cell 2	cell 1
1	r_{26}, r_{27}	$E_{m_{k+1}}, v_l$	α_p, d	v_l						v_l	α_p	$E_{m_{k+1}} v_l$
2	r_{28}, r_{35}	y	α_{p+1}, d	v_l, yes						v_l, yes	α_{p+1}, d	y
3	r'_1, r_{34}, r_{36}		α_{p+2}, d	y'		v_l, yes		v_l, yes		y'	α_{p+2}, d	
4	r'_7, r'_5, r_{37}		α_{p+3}, d		yes $v_l \notin T_1$	v_l, yes		v_l, yes	yes $v_l \notin T_2$		α_{p+3}, d	
5					y, v_l, yes		yes		y, v_l, yes			

 Table 4: Expanded Communication Configuration for $2\Delta^2$ and $1\text{-}\Delta^2$. Accepting Communication. Note that v_l , found in cell x are those which are elements of $T_1 \cap T_2$.

step	rule	cell 1	cell 2	cell 3	cell 4	cell x	$\langle 1, 0 \rangle$	0	$\langle 2, 0 \rangle$	cell x	cell 4	cell 3	cell 2	cell 1
1	r_{26} r_{27}	$E_{m_{k+1}}$ v_l	α_p d	v_l	ϵ_i γ_i v_l						ϵ_i γ_i v_l	v_l	α_p d	$E_{m_{k+1}}$ v_l
2	r_{26} r_{28} r_{35}	y	α_{p+1} d	v_l, yes	ϵ_i γ_i v_l						ϵ_i γ_i v_l	v_l, yes	α_{p+1} d	y
3	r_{26} r_{36} r'_1		α_{p+2} d	y'	ϵ_i γ_i v_l		v_l, yes		v_l, yes		ϵ_i γ_i v_l	y'	α_{p+2} d	
4	$r_{26}, r_{34},$ r_{38}, r_{37}		α_{p+3}, d		$v_l, \epsilon_i, \gamma_i$		y		v_l^2, yes^2 $y, v_l \notin T_1$ $\text{yes } y$		$v_l, \epsilon_i, \gamma_i$		α_{p+3}, d	
5	r_{26}, r_{39}		α_{p+4}, d		$v_l, \epsilon_i, \gamma_i$	y			v_l, yes^2	yes y v_l	v'_l, y', ϵ_i $\gamma_i, v_l'',$ $l'' \neq l$		α_{p+6}, d	
6	r_{26}, r'_2		α_{p+7}, d		$v_l, \epsilon_i, \gamma_i$	y			v_l, v'_l y', yes^2	yes y v_l	v'_l, y', ϵ_i $\gamma_i, v_l'',$ $l'' \neq l$		α_{p+7}, d	
7			α_{p+8}, d		$v_l, \epsilon_i, \gamma_i$	y	v'_l, yes	y', yes	v_l	yes y v_l	v'_l, y', ϵ_i $\gamma_i, v_l'',$ $l'' \neq l$		α_{p+7}, d	

Table 5: Expanded Communication Configuration for $1\Delta^3$. Accepting Communication.

step	rule	cell 1	cell 2	cell 3	cell 4	cell x	$\langle 2, 0 \rangle$	0	$\langle 3, 0 \rangle$	cell x	cell 4	cell 3	cell 2	cell 1
4	r_{26} r_{37} r_{38}		α_{p+3} d		ϵ_i γ_i v_l		$v_l^2 y$ yes ² v_l, yes		v_l yes y		ϵ_i γ_i v_l		α_{p+3} d	
5	r_{26} r_{39}		α_{p+4} d		ϵ_i γ_i v_l' $v_l,$ y		v_l yes ² v_l yes		v_l yes	y	ϵ_i γ_i v_l		α_{p+4} d	
6	r_{26} r'		α_{p+5} d		ϵ_i γ_i v_l		v_l' y' yes ²		v_l yes	y	ϵ_i γ_i v_l		α_{p+5} d	
7	r_{26} r_{40}		α_{p+6} d		ϵ_i γ_i v_l		y' yes		$v_l' \text{ yes}$ $v_l \text{ yes}$ $v_l' \text{ yes}$		ϵ_i γ_l v_l		α_{p+6} d	
8	r_{26} r_{39}		α_{p+7} d		ϵ_i γ_i v_l		y' yes		$v_l \text{ yes}$ $v_l' \text{ yes}$		$\epsilon_i v_l'$ $\gamma_i y'$ v_l		α_{p+7} d	
9	r_{26} r'		α_{p+8} d		ϵ_i γ_i v_l		y' yes		$v_l' y'$ $v_l \text{ yes}$ $v_l' \text{ yes}$	y	ϵ_i γ_i v_l		α_{p+8} d	
10			α_{p+8} d		ϵ_i γ_i v_l			y' yes	v_l' v_l $v_l' \text{ yes}$	y	ϵ_i γ_i v_l		α_{p+8} d	