Computation of decision problems within messages in DNA-tile-based molecular nanonetworks

Florian-Lennert Adrian Lau *, Florian Büther, Regine Geyer, Stefan Fischer

University of Luebeck (Institute of Telematics), Ratzeburger Allee 160, 23562 Lübeck, Germany

A B S T R A C T

Akyildiz et al. envisioned the use of nanonetworks as a new paradigm for computation and communication on a very small scale.

We present a new approach to implement nanonetworks with molecular communication using tile-based self-assembly systems on the basis of DNA. In this model, the medium of communication is filled with DNA-based molecules. Furthermore, some nanobots are capable of creating or releasing said molecules. Once present, they can be detected by other nanobots and interpreted as messages. Some DNA-based molecule systems are capable of universal computation. We show that it is possible to construct systems, in which the evaluation to TRUE of an arbitrary decision problem is a precondition for the assembly of a message molecule. We relocate computations from nanobots into message molecules, thereby revolutionizing the paradigm for computation in nanonetworks. This approach can be interpreted as computation inside the communication channel. We further present message molecules that only assemble if a marker has been detected at least k times, as a proof of concept.

© 2019 Elsevier B.V. All rights reserved.

1. DNA-based nanonetworks

DNA has been proposed as a material for construction as early as 1982 by Seeman [1]. Among others, this early work inspired and accelerated research that targets computational processes at the nanoscale. In recent years, nanonetworks have been introduced as a methodology to compensate for the resource constraints to be expected at increasingly small scales [2]. In nanonetworks, nanoscale devices collaborate to solve tasks that exceed the capabilities of single devices. To achieve this, the devices have to communicate with each other. For example, terahertz communication and molecular communication have both been examined as supportive technologies.

Over the past decade, numerous scenarios have been proposed in which the application of nanonetworks might be beneficial. Material sciences, medical, pharmaceuticals as well as computer science and chemistry are among the most prominent contributors. Medical applications appear to be most established. Since molecular communication (e.g. hormones) is native to living organisms, it is of special interest in medical scenarios.

In [3], a huge variety of applications for molecular communications have been surveyed. The detection and treatment of diseases are of general interest. Both can be further itemized into categories like drug delivery, immune system support, nanosurgery or the constant monitoring of health parameters.

In this work, we present a novel method for molecular communication in nanonetworks based on DNA-self-assembly systems [4,5]. Since DNA naturally occurs in the human body, DNA molecules are a prime candidate for molecular communication. The two main methods for construction with DNA are DNA-origami and tile-based self-assembly systems [6,7]. We focus on DNA-based technologies as DNA has high bio-compatibility since there are few antigens against DNA or RNA.

DNA-origami is an established, replicable technique which generates nanoscale objects (e.g., message molecules) with the desired physical properties [7]. The process utilizes a long strand of DNA that is pinned together at predetermined base pairings by a short DNA-strand. While it is possible to create complex structures using DNA-origami, the procedure is clearly limited in its capabilities. The longer the strands get, the higher the error-rate. Nevertheless, it has been shown that DNA-origami can create nanostructures that are capable of detecting certain markers and releasing a payload on detection [8].

We further investigate the tile-based self-assembly method that is also based on complex DNA-molecules. For this, we chose molecules that greatly differ from the conventional DNA double-helix that represents the genetic material of all organisms. An example is the DX-molecule that combines two parts of a double helix into a single molecule with open strands to both sides [9].

* Corresponding author.
  E-mail addresses: lau@itm.uni-luebeck.de (F.-L.A. Lau),
buether@itm.uni-luebeck.de (F. Büther), geyer@itm.uni-luebeck.de (R. Geyer),
fischer@itm.uni-luebeck.de (S. Fischer).

Those open strands can encode conditions, which have to be met for other DNA-molecules to form a stable binding.

If correctly arranged, DNA can be shaped into complex objects with multiple open ends in many directions. The molecules, often called *tiles* (see Fig. 1), can then be introduced into a common medium where they start to self-assemble into even more sophisticated objects. The process is mostly used for computational purposes or to create structures at the nanoscale [10].

In literature, it is often suggested that nanobots communicate through specific molecules [11]. We instantiate these *message molecules* through tile-assembly. A message molecule can either be a single tile or a more complex assembly of tiles. The process of tile-based self-assembly does not require nanobots to produce a complex molecule—a seed-tile is enough to initiate a self-assembly process to generate complex message molecules. Nanobots thus do not need to produce big molecules, simplifying nanobot design.

A message molecule may require a specific tile in order to finalize self-assembly. With all other tiles already present in the communication medium, a nanobot needs to release only a single tile as trigger to start or continue self-assembly. Tiles and assemblies of tiles (in this case message molecules) are subject to Brownian motion like any other molecule and may utilize it as a way of propagation. Every message molecule might be detected by other bots, which may amplify the message by producing/releasing additional tiles until a concentration threshold is met. Another option is the immediate release of therapeutics or fluorescence markers that indicate a positive diagnosis.

In the later presented scenario, the bots communicate among each other via self-assembly binding reactions of message molecules that might harbor binary information, as shown in Section 3.2. Communication to the macroscale may be realized by reading molecule concentrations, as with regular DNA-computing [12], or by reading information encoded on the message molecule with fluorescence markers.

Nanobots initiate communication with the production of only one, small type of tile. The required productions cost necessary might be greatly reduced. Other techniques require the production or release of complex and bigger molecules.

Another major benefit of tile-based message molecules is that computations are shifted from nanobots themselves into the assembly process of message molecules. This modifies the computation paradigm in nanonetworks, where it is generally assumed that nanobots execute computations. The space restrictions in the communication channel are far less severe. To the best of our knowledge, to this day, no major research explores the possibility of computations taking place in the communication channel.

Any decision problem [13] can be computed by a tile-assembly system as shown by [6]. However, it remains unclear if nanobots may perform computations like classical computers, or fulfill the space requirements imposed by complex computations [14]. Thus, it appears beneficial to utilize a well tested process to encode conditional behavior at the nanoscale, thereby answering the question of how computations may look like in nanonetworks.

The remainder of this work is structured as follows: Section 2 explains the necessary basics for tile-based self-assembly systems. Section 3 elaborates the use of tile-based self-assembly systems for the purpose of molecular communication and explains the relevant basic components. Section 4 compares the presented framework to other channel models and evaluates the approach. Section 5 concludes the work, and Section 5.1 presents ideas on future work and some open questions.

2. Preliminaries for tile-based self-assembly

This section provides definitions and examples of the mathematical constructs known as *tiles* and *assemblies*, as well as all relevant properties [6]. The definitions are based on [15]. For an in-depth introduction to tile-based self-assembly see [12]. Assemblies, made up out of tiles, serve as messenger molecules in this communication scheme, and can additionally compute arbitrary problems through self-assembly.

Definition 1. An *n-dimensional tile* (t_n) is an object in the $\mathbb{Z}^n$ with unit length and angles of 90°. DNA-tiles cannot be rotated.

A *side* of a tile t_n is defined by a vector $u_t \in \mathbb{Z}^n$. u_t has exactly one non-zero element with value 1. Sides are enumerated by:

$$\text{side} : t_n \mapsto \underbrace{U_t \times U_t \cdots \times U_t}_{n}$$

U_t is the set of unique unit vectors u_t defining the sides of t_n. The vector $-u_t$ denotes the side opposite of u_t in the same dimension.

In the following, we often omit the dimension of tiles when this information is irrelevant in the presented context. We focus on two- and three-dimensional tiles. A 2D-tile t_2 is a square in the $\mathbb{Z}^2$ and a 3D-tile t_3 is a cube in the $\mathbb{Z}^3$. Fig. 2 shows an example of a 2D-tile.

Each side of a tile can have one or multiple *glues*:

Definition 2. A *glue* $g \in G$, where G is the set of utilized glues, is defined by a *label* $\mathcal{L}_g \in \Sigma^*$, where Σ is an alphabet and $s \in \mathbb{N}$ a glue-*strength*, as well as the functions:

$$\text{label} : G \mapsto \Sigma^*$$
$$\text{strength} : G \mapsto \mathbb{N}$$

The glue label is often referred to as *color*. Glues are instantiated by open DNA strands that can encode both conditional bindings and varying strengths.

In the DNA-computing community, tiles are implemented with DNA [12]. An example structure can be seen in Fig. 1. The open strands can be used to create a variety of different glues.

Each tile can occupy exactly one position in the discrete space $\mathbb{Z}^n$, given by the vector $v_t \in \mathbb{Z}^n$. Such a position is called a *site*.

Definition 3. Two tiles t and t' that occupy sites defined by the vectors v_t and $v_{t'} \in \mathbb{Z}^n$ are *neighbors* iff $|v_t - v_{t'}| = 1$ and the resulting vector $e = v_t - v_{t'}$ has exactly one non-zero element.

A tile can have zero or exactly one neighbor on any of its sides. There is no interaction between the tiles t and t' unless they are neighbors. The rules for this interaction are specified by their glues.

Definition 4. An *n-dimensional tiletype* T_n is a template for a tile t_n. It is defined by a label $\mathcal{L}_T \in \Sigma^*$ and a set of glues $g_i \in G$, one on each side $u_{t,i} \in U_t$ of t_n, and the functions:

$$\text{glue} : T_n \times U_t \mapsto G$$
$$\text{strength} : g \in G \mapsto \mathbb{N}$$

All tiles that have identical glues and the same markers are of the same tiletype.

All not explicitly mentioned or shown glues have the label $\mathcal{L}_g = \text{VOID}$ and strength $s = 0$.

A tile t that has a glue of strength zero or no glues on side u_t cannot interact with its neighbor t' at the side u_t.
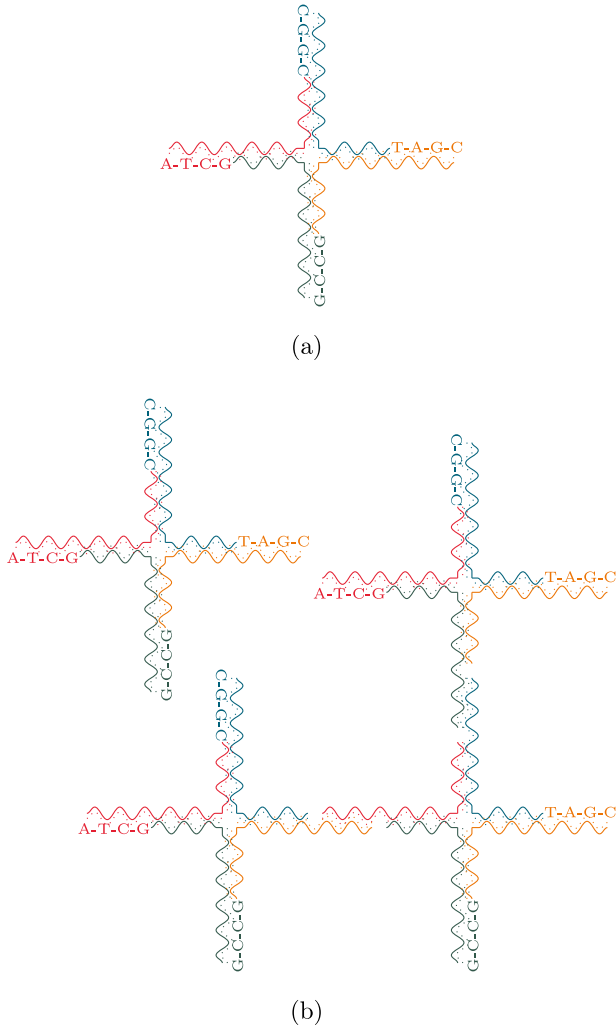
(a)

(b)

Fig. 1. (a) A schematic representation of a DNA tile with one sticky end in every direction. (b) Three DNA tiles binding together and a forth tile in the process of binding to the assembly.
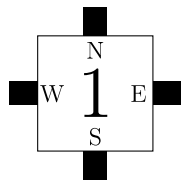

Fig. 2. A 2D-tile with a binary marker in the middle and a glue on each side colored {N, E, S, W}.

Definition 5. The *temperature* τ of a tile system describes the minimum glue strength required to form a stable binding.

Two tiles can *bind* together if they are neighbors and have glues of matching color ("matching" with DNA-tiles means that both tiles have complementary open strands) as well as a strength of at least the temperature τ.

The temperature is modeled after the physical temperature. As the physical temperature increases, molecules move faster, thus increasing the chance of molecular bonds to break. Molecules with stable and stronger bonds are more stable at higher temperatures than those with fewer or weaker bonds.

Definition 6. Two tiles t, t' of type T and T' *correctly bind* at temperature τ if:

(1) t and t' are neighbors

(2) $\exists\, u_i \in U_t : \text{label}(\text{glue}(t, u_i)) = \text{label}(\text{glue}(t', -u_i))$

$\wedge\ \text{strength}(\text{glue}(t, u_i)) \geq \tau$

$\wedge\ \text{strength}(\text{glue}(t', -u_i)) \geq \tau$

Bindings that do not fulfill the conditions (1) and (2) are considered *incorrect* or *errors*.

Due to the nature of DNA, some bindings will always be incorrect [6]. However, the number of incorrect bindings can be reduced by designing special sets of tiletypes that account for error behavior. [12] shows two approaches to mitigate errors for two-dimensional assemblies.

The *binding strength s* between the tiles t and t' is equal to the strength of the matching glues on the involved sides. The strength of a glue is denoted by the number of black bars on a side of the tile. In Fig. 2, all glues are of strength 1.

The *total binding strength* of a tile t is the sum of the binding strengths between t and all of its neighbors. If a glue does not have matching labels with a neighboring glue it is a *mismatched glue*. Mismatched glues do not increase the total binding strength. A tile that does not meet the minimum total strength requirement of τ or has a mismatched glue, is a *mismatched tile*.

Definition 7. An *n*-dimensional *tile assembly*, or just *assembly*, is a partial function $\alpha : \mathbb{Z}^n \mapsto T$, where T is a set of *n*-dimensional tiletypes T_n. An assembly α is τ-stable if no tile $t_n \in T$ can be removed without breaking bonds of at least strength τ, where $\tau \in \mathbb{N}$.

Definition 8. The *border* of an assembly α is a subset of tiles of α. It includes all tiles that have an unoccupied site, that is, where the tile has no neighbor.

Only border tiles can interact with tiles added to the assembly at a future time.

Definition 9. The *growth front* of a *n*-dimensional assembly is a subset of all sites in the $\mathbb{Z}^n$. A site belongs to the growth front iff it is not occupied by a tile and a neighboring site is occupied by a border tile with a minimum glue-strength of 1 towards the empty site.

The sites of the growth front may change when a tile is added to or removed from the assembly. For the sake of simplicity, we assume that a single random tile is added to the assembly α at every discrete time step.

The initial assembly α_0 at time step 0 is the *seed-tile* or *seed assembly* σ. Starting from σ, one tile is non-deterministically added to α at a growth front site at each time step. Due to the non-deterministic nature, it can be complicated to design tile sets that assemble into a unique terminal assembly.

Definition 10. A *Tile Assembly Model* (*TAM*) is a three-tuple $\mathcal{T}_\tau = (T, \sigma, \tau)$, where T is a finite set of tiletypes, σ a seed assembly and $\tau \in \mathbb{N}$ the *temperature* of the TAM.

In the remainder of the text τ is assumed to be 2 and the stability of a TAM is omitted.

Definition 11. $\mathcal{A}[\mathcal{T}]$ denotes the set of all terminal assemblies that can be produced by a TAM $\mathcal{T}$ in a finite number of time steps. An assembly $\alpha \in \mathcal{A}[\mathcal{T}]$ is *terminal* if no tile can be τ-stably added to it.
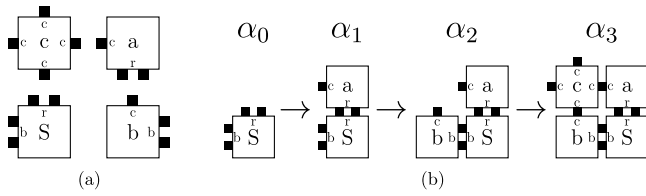
Fig. 3. (a) An example tile set of a 2D-TAM as a visual representation. (b) The tile assembly sequence of a TAM with a tile set as depicted in (a). The seed tile is S and temperature $\tau = 2$. $\langle \alpha_0, \ldots, \alpha_2 \rangle$ depict the three time steps the TAM requires to reach a terminal assembly.

Definition 12. Let α_i be an assembly produced by a TAM $\mathcal{T}$. An *assembly sequence* of TAM $\mathcal{T}$ is the sequence $A = \langle \alpha_0, \alpha_1, \ldots \rangle$, where α_{i+1} is obtained from α_i by adding one tile to α_i. If the sequence A is finite, the last element of the sequence A is the *result* of $\mathcal{T}$.

Fig. 3(a) shows an example tile set of a 2D-TAM, where the seed is the tile labeled S. The assembly sequence of this TAM with temperature $\tau = 2$ is depicted in Fig. 3(b). α_0 is the initial assembly which consists of only the seed tile. α_1 and α_2 show the addition of one tile each to the previous assembly.

A *tile-assembly system* (TAS) is an extension of a TAM. It governs the rules at which and at what pace tiles are added to or removed from a growing assembly. There are several prominent tile-assembly systems. The most popular ones date back to Eric Winfree [6] and are the abstract Tile-Assembly Model (aTAM) and the kinetic Tile-Assembly Model (kTAM) [12]. All tileset verifications in this work use the realistic kTAM model. It reflects the nature of DNA by introducing a probability that tiles may break free from the assembly.

2.1. Errors in self-assembly systems

There are three types of errors commonly observed in tile-based self-assembly systems like the kTAM [12].

Definition 13. A *growth-error* occurs when a tile is attached to the assembly in such a way that at least one of its glues mismatches an adjacent glue.

Definition 14. A *facet-error* occurs when a tile is attached to the assembly without any mismatched glues but with insufficient binding strength to satisfy the temperature requirement.

Both error types can become *locked in*, when a new tile is attached at a neighboring site, providing correct bonds to the erroneous tile. Therefore, the erroneous tile's total binding strength may satisfy the temperature requirement and thus be unlikely to detach from the assembly. Fig. 4 depicts the process. Growth-errors and facet-errors can be significantly reduced by employing block-replacement strategies like *snaked-proofreading* or $k \times k$-*proofreading* [16,17].

The third kind of error differs from the other two. It occurs, when the assembly process starts with tiles other than a seed tile.

Definition 15. A *nucleation-error* occurs, when a tile-assembly system starts an assembly process with no seed tile present or with any other tile than the seed.

3. Tile-based communication systems

This section introduces the most important components of tile-based molecular communication systems, namely senders, channels and receivers as well as messages and a mechanism for debugging. Furthermore, it details a construction scheme for a message molecule that computes a logical 4-bit AND.
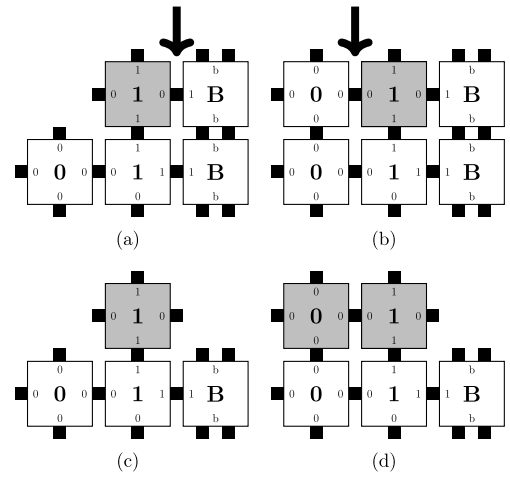

Fig. 4. (a) A growth-error at the gray tile. (b) The locked growth-error. (c) A facet-error at the gray tile. (d) The locked facet-error.

3.1. Nanonetworks and nanobots

This subsection briefly defines the types of nanobots that exchange information in the following scenarios. We use slightly modified definitions from [18].

Definition 16. A *nanobot* is an artificial construct with an overall nanoscale size, designed to perform a predefined function in an environment Γ. In the scope of this work, it consists of a power supply P, a sensor/receptor S, an actuator/dispenser A and means of communication C.

Nanosensors are similar to nanobots, but do not possess an actuator component A.

Definition 17. A *nanonetwork* is a directed ad-hoc graph $G = (V, E)$, where V is a set of *nanobots and nanosensors*, operating in an environment Γ. E is a set of Tuples $(V \times V)$, or *edges*, that describes the connections between the nanobots.

Nanosensors and nanobots form a *nanonetwork* that communicates and computes – in the presented approach – via self-assembled message molecules.

3.2. Message molecules

Message molecules are assemblies of tiles moving via Brownian motion through a medium/channel, e.g., a petri dish. Other nanobots can detect the ligands of the molecular message. Generally, it would suffice to use just a single tile to transfer information, as the class of assemblies we analyze always assembles into exactly one deterministic object. Therefore, DNA-tiles could be used to implement regular molecular communication.

Since we propose to execute simple computations "inside" of message molecules, we require more than a single tile. With this approach we modify the computational paradigm at the nanoscale, where it is generally assumed that computations happen within nanobots.

Fig. 5(a) depicts the set of tiletypes for a 4-bit AND message molecule. The colors encode different semantic properties. Yellow tiles are receptor/ligand-tiles—a receptor or ligand is a part of a terminal assembly with unbound glues. Green tiles are inhibitor-tiles that lower the chance of erroneous bindings at the assembly border. They also increase the assembly's stability. Furthermore, an extra set of tiletypes of only inhibitor-tiles can be designed for
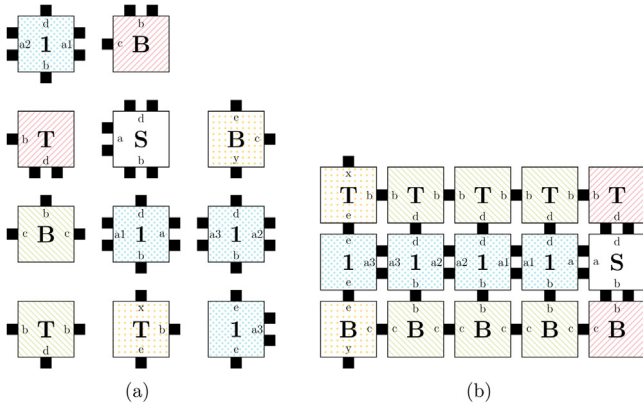
Fig. 5. (a) A tileset resulting in a 4-bit AND message molecule at temperature 2. S is the seed. (b) The resulting message molecule. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
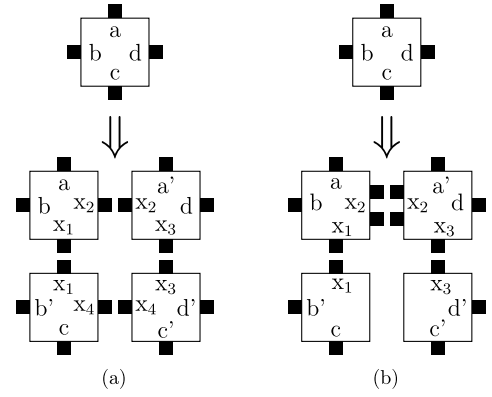

Fig. 6. (a) The tiletype is replaced by a 2 × 2 block of tiletypes. (b) The tiletype is replaced by a 2 × 2 block of snaked proofreading tiletypes.
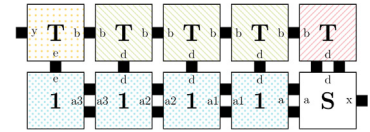

Fig. 7. An adapted message molecule at temperature 2 that mitigates nucleation-errors.

every glue in a TAM to globally suppress assembly. Blue tiles are *conditional-tiles* that actually calculate binary decision problems. Note, that the marker in the middle of the conditional tile does not represent its binary value—the presence of a tile represents a logical TRUE. The white tile is the seed-tile that starts the assembly and the red tiles enable growth in the *y*-dimension.

Fig. 5(b) shows the resulting message molecule. This assembly fulfills a multitude of required properties. It consists of a seed-tile to the right labeled "S". The assembly advances by adding a border tile to the top and the bottom of the seed-tile. The tiles at the border of the assembly have as few glues to the outside as possible, to prevent erroneous bindings. The four tiles to the left of the seed-assembly are conditional-tiles. In this case, they encode the computation of a 4-bit logical AND.

The message molecule requires the presence of all tiles – especially the four different conditional-tiles – to fully assemble. Releasing conditional-tiles only under special conditions (e.g. in the presence of a disease-related biomarker) allows for the encoding of binary information into conditional tiles. By adding additional distinct conditional tiles, one can easily compute a *k*-bit AND. This approach can be utilized to find a *distributed consensus* concerning certain events among nanodevices. By employing this strategy, the number of false positives can be greatly reduced.

3.3. Assembly errors in message molecules

Message molecules are prone to the three types of errors presented in Section 2.1. Facet-errors and growth-errors can both be reduced significantly by employing block-replacement schemes like *snaked-proofreading* or *k×k-proofreading* [16,17]. Fig. 6 shows an example. $k \times k$-proofreading simply increases the scale of the assembly, thereby rendering growth-error $1/k$ times as likely. In addition, snaked-proofreading also prevents facet-errors in one direction completely, by omitting the internal glue between the two lower tiles of the snaked-block. By replacing each tiletype by a block of tiletypes, more errors are required for a whole original tile to be erroneous. The in-message calculation remains equivalent.

Nucleation-errors are a different matter. In theory, the leftmost two columns of the message molecule in Fig. 5 can form without the presence of a seed-tile, thereby forming a ligand without a successful computation. Two facet errors combined with a nucleation-error are sufficient to produce this behavior. The probability for a single facet-error – namely two tiles attaching next to each other with insufficient strength *at once*, thereby

locking each other in place – is very low. Two of them happening at the same nucleation-error is even more unlikely. Due to the high concentration of tiles, the described scenario will eventually occur, however unlikely.

This can be circumvented by the presented snaked block-replacement strategy. However, it is necessary to adapt the outer glue structure to ensure that only two outer glues are present.

Another possibility would be a redesign of both receptor and message molecule to also enforce the presence of the seed-tile. Fig. 7 shows a possible adjustment. An appropriate receptor can easily be designed. This way, the presence of both the seed-tile and the ligand are enforced and therefore the effects of nucleation-errors are mitigated.

3.4. Message receivers

Fig. 8 shows two examples of possible receptors for message molecules. Both receptors and ligands are based on DNA. The binding of a ligand at a receptor is a DNA binding reaction that is subject to the temperature τ.

The gray squares represent the arbitrary ligand/receptor-part of the message molecules. The black squares represent individual glues of strength 1. In order for a message molecule to form a stable binding with a nanobot, at least a binding strength equal to the system temperature τ is necessary.

If designed accordingly, nanobots can detect the last three tiles that were added to a message *only at once*. This is necessary in order to prohibit the binding of single tiles with the receptors and thus reducing the probability of binding errors at receptors.

Once a message molecule binds at a receptor, this might trigger any kind of reaction. This could, e.g., be an own release of seed-tiles and thereby help spread messages through the medium. A single nanobot is not necessarily limited to one type of receptor.

If the goal is just communication, single tiles are enough to transmit binary information. Assemblies of multiple tiles are only required if the message should perform additional computation.
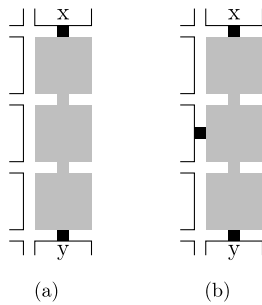
Fig. 8. A receptor assembly designed to bind message molecules at temperature 2 (a) with strength 2 and (b) with binding strength 3.

3.5. Example for in-message computation

We envision the application of DNA-tile based nanonetworks in the blood stream of living organisms. Since there are numerous unknown environmental parameters in the human body – in addition to the technical challenges – we illustrate the usefulness of the presented approach in a more constrained, but easily achievable, realistic scenario.

To quickly and reliably detect a disease, a DNA-tile based nanonetwork can be used in a controlled environment like a petri dish. The components for the nanonetwork and a blood or tissue sample can be mixed in a petri dish, where it is easier to control the environmental parameters that are required for self-assembly systems.

If a disease has been detected, conditional tiles are released and distributed via Brownian motion. In a petri dish, the tile concentration can be chosen as high as necessary to reliably assemble message molecules. Additionally, we extrapolate from wet lab experiments that successfully assembled a Sierpinski triangle with a tile-based approach.

Once the molecules are fully assembled, they behave like any other molecule that acts as message carrier in a molecular communication system and can transmit the information to a receiving nanobot, that in turn releases a chemical that starts a fluorescence reaction. This scenario illustrates how nanonetworks could be used to help diagnose diseases with a high level of confidence.

We design a tile-based nanonetwork consisting of a number of nanobots and sensors that dispatch specific tiles once a marker is detected. To ensure that few "false positive" messages occur, our tile-assembly based nanonetwork guarantees that at least 4 individual nanobots detected a marker for a whole message molecule, with ligands, to assemble. We utilize the set of tiletypes from Fig. 5 for this scenario.

We include the computation of a 4-bit logical AND into our message molecules, thus proposing a method to immediately release therapeutics or fluorescence markers once a disease is reliably detected. The 4-bit AND ensures a distributed consensus about an event with high probability.

Generally speaking, tile-assembly systems are Turing complete at temperature 2 or higher [6,13], thus similar systems can compute arbitrary decision problems and then add appropriate ligands that can be detected by other nanobots to identify a positive computational result.

Fig. 9 illustrates the vision: 1. Nanosensors detect the disease-related biomarkers (orange squares) and 2. release a number of conditional tiles 1–4 in return. 3. All other tiles constantly circle the channel (blood stream/petri dish) and start to further assemble once the conditional tiles are present. When all four tiles are gathered, the ligand-tiles can stably attach to the message molecule. 4. With the ligand function now present, other
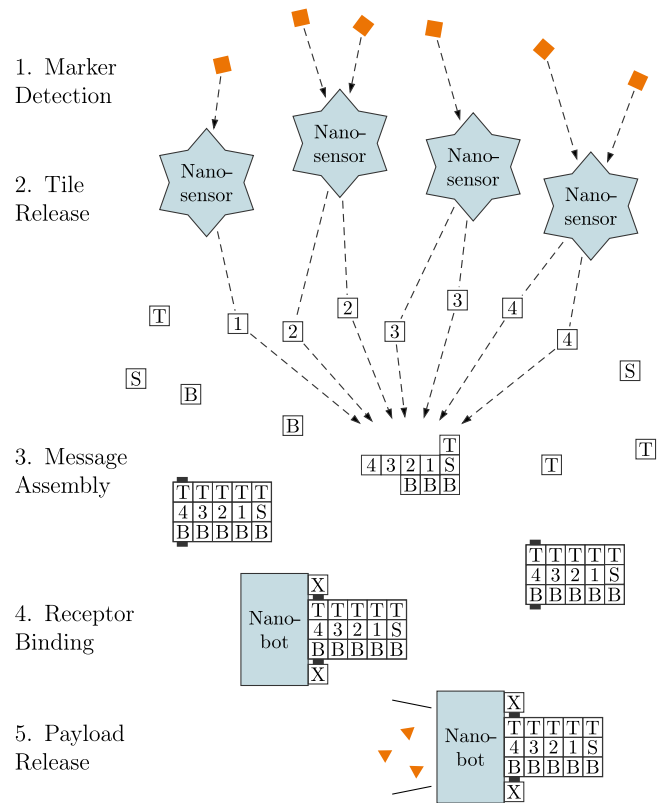

Fig. 9. Reliably recognizing disease-related biomarkers with a nanonetwork by computing an in-message 4-bit AND.

nanobots can bind the message molecule at their corresponding receptors, thus communicating the information that a disease has been reliably detected. 5. Nanobots release an antigen to help combat the detected disease.

The nanobots only detect a message molecule once it is fully assembled, and a message molecule only fully assembles once all required tiles are present. The set of tiletypes is designed in a way that it always assembles into exactly one terminal assembly while erroneous bindings are rare. All presented sets of tiletypes were tested and verified in a kTAM simulator [19,20].

3.6. Scenario modularization

The presented medical scenario serves to clarify the vision of nanonetworks. In reality, the experiments will start with much simpler scenarios in vitro. To test such a scenario in wet lab experiments, past experiences have shown that it can be beneficial to decompose a vision into smaller parts. Those components are then tested on their own, thus validating parts of the approach at lower cost.

Such a wet lab experiment could be realized by using a piece of human tissue in a petri dish and then testing the nanonetwork components separately. As a last step, all functioning parts have to be re-assembled into the scenario. A non-medical testing approach for the whole scenario, after parts have been tested, could be to use nanonetworks to monitor and manage agriculture or fish tanks.

The postulated scenario is rather generic. The phases are depicted in Fig. 9. The scenario can be generalized to the detection of an arbitrary marker. In essence, the scenario consists of the following parts:

1. Detection of a generic marker by a nanosensor. Li et al. proved in a wet lab experiment that therapeutics can be released by *detecting a specific molecule* [8,21].
2. Storing therapeutics, tiles or other payload in/on a nanobot. A variation of this problem has even been solved in vivo by Li et al. [8,22].
3. Assembling the 4-bit AND message molecule under laboratory conditions. In general, this should be possible, since many experiments in the DNA-computing community yield a lot more complex structures [23].
4. Attaching a fully assembled message molecule to a pre-assembled receptor. In theory, this should also be possible. The DNA-computing community created the so-called 2HAM model that simulates the interaction between partial assemblies. The process seems to work under laboratory conditions [24]. A receptor may be assembled using a similar approach as in the message molecule. A receptor can therefore be an assembly.
5. Release of therapeutics after a message molecule has been detected at a nanodevice. Again, Li et al. solved variations of this problem [8].

These partial problems have mostly been solved using a DNA-based approach. To the best of our knowledge, reliable molecule release mechanisms remain an open problem in tile-based self-assembly systems. It would be necessary to locally decrease the stability of a nanobot or molecule to open it and release a payload. In theory, DNA-tiles are compatible with other DNA-based approaches and it should thus be possible to adapt already functioning technologies, like the mechanical opening mechanism presented by Li et al. [8].

4. Evaluation

Since no wet lab experiments have yet analyzed communication in tile-based nanonetworks, we mainly investigate theoretical channel models for molecular communication with a coherent biological basis. We evaluate our proposed scenario by both simulation and comparison with other molecular communication approaches—which is only partially possible. Further, we extrapolate from previously conducted wet lab experiments to further strengthen our reasoning.

4.1. Simulation and validation

We simulated the presented scenario using the software ISU-TAS by Patitz [20]. We instantiated the tilesets for both the kTAM and the 2HAM model. The kTAM realistically simulates the assembly of exactly one message molecule. The model accurately predicts the assembly of tiles as validated in [25]. We used the 2HAM to validate the interaction between receptors and message molecules and analyzed the interactions between intermediate assembly products. Once the message molecules are fully assembled, we expect them to behave like regular molecules used for communication. We refer to established molecular communication models to further strengthen our line of argumentation.

Fig. 10 shows the tileset for both the kTAM and the 2HAM simulation. The blue tiles model the receptor. The remaining tiles assemble into a message molecule.

KTAM. We simulated the assembly of the message molecule with parameters $G_{se} = 10.4$ and $G_{mc} = 17.0$—the same values as in [25]. G_{se} describes the free energy cost of breaking a single bond and G_{mc} the monomer concentration. The ratio of the binding cost to the strength of a single bond $\tau = \frac{G_{mc}}{G_{se}}$ represents the temperature parameter of the kTAM.
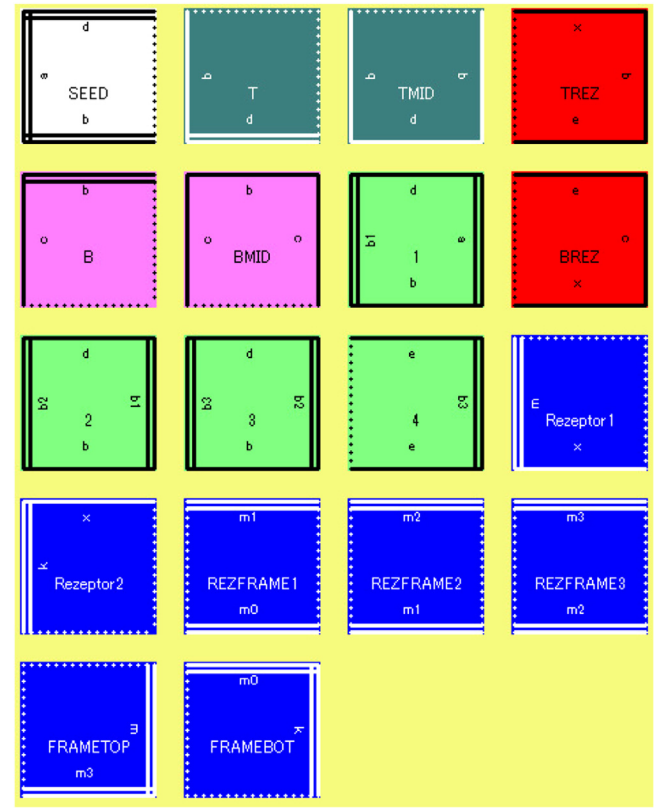

Fig. 10. The tileset for both the receptor (blue tiletypes) and the message molecule. The text in the center of a tile represents its label. The number of lines on the edge of a tiletype represents the strength of a glue. The color of a glue is denoted by a label next to it. A dotted line means that a glue has strength 0. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

A value of $G_{se} = 10.4$ led to more errors in the simulation result in [25], we therefore simulated under problematic conditions. A $G_{mc} = 17$ value corresponds to 0.8 µM per tile and represents the monomer concentration. A $G_{se} = 10.4$ value corresponds to a temperature of 32.7 °C. Lower values of G_{se} represent a higher temperature, e.g., $G_{se} = 8.5$ corresponds to 41.8 °C.

The simulator assumes that every tiletype occurs equally often in the medium and sufficient tiles for additional growth are always available—this includes tiles for the receptor, which slows down assembly. In reality, the individual tile concentrations can be manipulated to further enable control over the growth process.

The simulation showed that a message molecule fully assembled on average after approximately 3000 attempted binding reactions. Ideally, only 28 binding reactions are necessary for the assembly. Fig. 11 shows the result of the simulation.

Fig. 12 shows a histogram of the results of 100 assembly simulations. The number of binding reactions (both forming and breaking bonds) necessary to fully assemble a message appears to be Log-normal distributed, with a slight bias towards higher numbers ($\sigma \approx 0.5$). Only one fully assembled message occurred in under 1500 reactions. In reality, numerous binding reactions happen in parallel and the required number of binding reactions is an indicator of the time required to fully assemble a message. As long as $2G_{se} > G_{mc} > G_{se}$ holds, tile associations are more likely than tile disassociations and a fully assembled message molecule will eventually form [26].

No simulation run showed an erroneous interaction between the message molecule and the tiles that model the receptor. Even after more than 100 000 binding reactions, the fully assembled message showed no errors and remained stable.
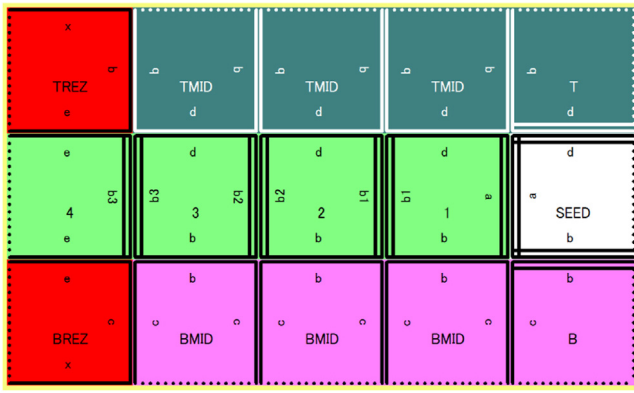
Fig. 11. The fully assembled message molecule from a kTAM simulation with ISUTAS at $G_{se} = 10.4$ and $G_{mc} = 17.0$ after 3000 binding reactions.
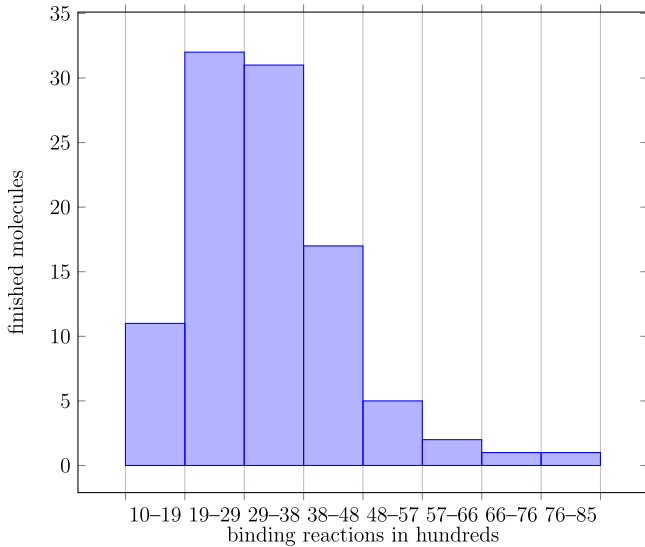

Fig. 12. Result of 100 message molecule assembly simulations. The histogram shows the number of finished molecules with their respective number of binding reactions.

Table 1
Size and number of possible partial assemblies and the final assembly that may occur simultaneously when many seeds are present at once.

Size	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	22
Number	18	12	11	11	11	12	11	11	10	9	7	5	3	2	1	1

2HAM. After validating that the message molecule realistically self-assembles, we further modeled the binding process between receptor and message molecule with the 2HAM model from ISU-TAS, as well as the assembly of both the receptor and the message molecule. The blue tiles in Fig. 10 depict the receptor. The remaining tiles assemble into a message molecule. As displayed in Table 1, numerous intermediate assemblies exist for a tile-assembly system that consists of receptors and message molecules. Since many assemblies form at once, it should not be possible for partial assemblies to interact with each other.

The intermediate assemblies cannot interact with each other to form stable molecules, unless multiple errors occur at once. Only when both the receptor and the fully assembled message molecule are present, a receptor binding may occur. Fig. 13 shows the binding message molecule.
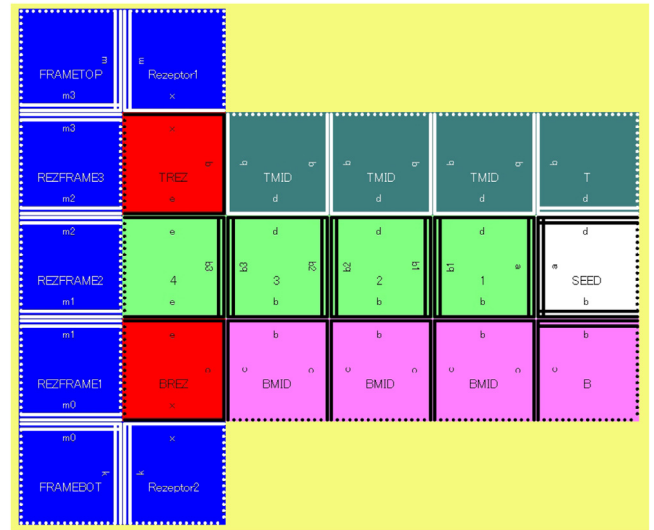

Fig. 13. The message molecule (right) binding at the receptor (left).

Sierpinski triangle experiment. [25] shows the result of a wet lab experiment with DNA-tiles. In reality, DNA-based tiles assemble like a hybrid of both kTAM and 2HAM. The assembled Sierpinski triangle occurs multiple times in the medium. Apart from that, the assembly process behaves the same as proposed in our scenario. A tile concentration of 0.2 μM for every tile has been used, which is 1/4 of the simulated concentration. For comparison, the sodium levels in one liter of human blood are 0.14 μM. The results show that structures with up to 100 tiles can self-assemble with low error rates.

Since the proposed message molecule consists of only 15 tiles, it appears likely that it self-assembles without errors. Extrapolating from the results of [25], we propose to use a tile concentration of 0.2 μM for every tiletype.

Derived parameters. A comparison between the simulation and the physical experiments is difficult. At a tile concentration of 0.8 μM each tile, the simulator completed a message molecule after on average 3000 steps. The wet lab experiment was terminated after one hour at a tile concentration of 0.2 μM for each tile. The simulated message molecule contained 15 tiles, while the resulting assemblies of the wet lab experiment contained on average 100 tiles. By extrapolating from these results, we assume that it would take at most 10 min to fully assemble a message molecule of size 15. In general, bigger assemblies grow faster due to the bigger border.

For the number of required nanosensors and nanobots, a lower bound can be given. To ensure that for every seed-tile a message molecule may assemble, we have to guarantee that every conditional-tile occurs at least as often in the medium/the nanosensors. If placed in a DNA-box of size $42 \times 36 \times 36$ nm^3 [22], 472.5 conditional tiles of size $14.4 \times 4 \times 2$ nm^3 [25] can fit into the volume of a box. Equally distributed, 1.275×10^{10} DNA boxes would be necessary to harbor the amount of tiles used in the experiment with the volume of 50 μl [25].

In theory, we can freely choose how to distribute the conditional tiles among the nanosensors. In [25], the amount of basic components were overestimated a little to ensure proper assembly of the tiles—therefore we propose to do the same.

In summary, we expect the assembly of message molecules and receptors to properly function in a controlled environment like a petri dish. Given a high enough concentration of message molecules – which can be chosen appropriately – we expect the

Table 2

Comparison of molecular communication methods, adopted from [29]. k is the number of particles in one message, n is the number of different particle types available.

Based on	Propagation method	Bio invasiveness	Bits/Particle
Tiles	Diffusion	Very low	$1/k$
Particle number	Diffusion	High	$1/k$
Concentration	Diffusion	High	$1/k$
Particle type	Diffusion	High	$1/\log_2 n$
Particle order	Diffusion	High	$2/n$
Bacteria [30]	Chemotaxis	Low–high	Several
Virus particle [31]	Diffusion	Low–high	Several

molecular messages to assemble. In the presented scenario, a high particle density is of little concern.

The binding reaction between receptor and message molecule works like other molecular communication approaches that are based on Brownian motion with or without drift. We refer to [27] for a detailed analysis and a case study on molecular communication in e.g., the blood stream.

4.2. Information exchange efficiency

Table 2 displays a list of molecular communication categories and their corresponding capabilities. We evaluate the approaches with regard to underlying principle, propagation method, biological invasiveness and bandwidth in bits per particle. Since message molecules constitute of tiles, they can be arbitrarily big. Each tile can encode at least 1 Bit of information.

Table 2 shows that most techniques employ diffusion as a method for propagation. Only bacteria-based approaches differ, as they are capable of chemotaxis. Particle-based approaches usually use endogenous molecules, which can interfere with the body's own communication. Tiles and some bacteria probably will not conflict. We thus consider them to be less bio-invasive.

Tile-based molecular communication exhibits additional properties, which to the best of our knowledge are not possible with other approaches:

1. Message molecules can compute decision problems [6].
2. Message molecules can transport binary information in a constructive manner, e.g., by labeling parts of a tile with fluorescence markers.
3. Nanonetworks with tile-based message molecules can employ error correction for in-message computation [28].

Nevertheless, it is difficult to compare parts of our approach with other communication models. The hierarchical process of message molecule assembly is simply not represented. If only fully assembled message molecules are considered, they can be simulated the same way as molecular communication approaches that are based on particle types.

5. Conclusion

DNA-based molecular communication in nanonetworks is a bio-compatible way of exchanging messages within nanonetworks ex vivo. When message molecules comprise of DNA-tiles, they can incorporate computation capabilities into the messages. This in turn eases the computational burden on single nanobots, which become easier to design and construct.

This work introduced a modeling framework for molecular communication with self-assembling message molecules built from DNA-tiles. By drawing tiles from the ambient medium, these messages assemble in specific patterns. The exact assembly result depends on the seed and conditional tiles released from nanobots. A message may not fully assemble unless a logical condition is met. In the example scenario, four different nanobots have to release a conditional tile each in order to assemble a message, thereby implementing a distributed consensus. More generally, a message can compute arbitrarily complex decision problems (for example, AND, OR, XOR, MAJORITY, THRESHOLD…) and only fully assemble if these evaluate to TRUE.

The self-assembly property of DNA-tiles yields a constructive and reproducible implementation. The resulting message molecules possess a huge variety of structure and binding properties, while maintaining the same size. Careful design of the comprising tiles and terminal messages can reduce assembly errors, for example by adding tiles without glues to the outside of a message.

While on average approximately 3000 binding reactions are necessary to fully assemble a message molecule, this fact is of little concern in experiments with DNA. The monomer concentration can be increased until satisfactory results are met. As long as $2G_{se} > G_{mc} > G_{se}$ holds, tile associations are more likely than tile disassociations and a fully assembled message molecule will eventually form, given enough time [26].

DNA-tile based message molecules are naturally bio-compatible as well as bio-degradable. If applied in a living organism, the lifetime of message molecules is naturally limited by filter organs that influence DNA. Furthermore, very few antigens against DNA or RNA exist. The construction of DNA-tiles and DNA-origami is already possible in laboratories. Both are well tested, and can probably be combined. As such, DNA-tiles are a complete scheme for computing and molecular communication, which is implementable today.

Assembled message molecules may be inspected by outside observers. For example, some tiles can receive fluorescent labels, whose combination within a molecule can be detected from outside. A controller may even further influence or stop the assembly process by adding inhibitor-tiles or controlling the ambient temperature.

In summary, we presented and evaluated a scenario in which a tile-based nanonetwork can be applied to diagnose diseases ex vivo with high confidence. This is possible, since the involved nanobots calculate a distributed consensus about the presence of markers by a 4-bit AND.

5.1. Future work and open questions

The example scenario for a 4-bit AND message breaks down into five major parts. Each part can be verified in wet lab experiments on their own. Experiments regarding construction of message molecules and their interaction with receptors have already been conducted. Other parts, especially reliable tile dispensation, will need further investigation.

In this work, we presented only simple logical functions using tile-based self-assembly systems. Future work should analyze more complex approaches that have been proposed to be utilized in medical scenarios. We want to investigate how to combine the basic tilesets for logical operators into more complex logical formulas, and the implications for a tile-based messaging scheme under realistic conditions.

In a last step, we want to test the whole proposed scenario in a wet lab experiment, preferably in a petri dish and later in vivo.

In addition to wet lab experiments, it would be beneficial to simulate the presented scenario as accurate as possible. The human blood stream is of special interest. Previous experiments have shown the general feasibility of the approach ex vivo. For this, a complete model of the communication channel with the most important environmental parameters is necessary. Those include flow effects in the channel, collision detection, binding reactions in a fluid medium and tile diffusion.

References

[1] Nadrian C. Seeman, Nucleic acid junctions and lattices, J. Theoret. Biol. 99 (2) (1982) 237–247, http://dx.doi.org/10.1016/0022-5193(82)90002-9.

[2] Ian F. Akyildiz, Fernando Brunetti, Cristina Blázquez, Nanonetworks: A new communication paradigm, Comput. Netw. 52 (12) (2008) 2260–2279.

[3] L. Felicetti, M. Femminella, G. Reali, P. Liò, Applications of molecular communications to medicine: A survey, Nano Commun. Netw. 7 (2016) 27–45, http://dx.doi.org/10.1016/j.nancom.2015.08.004.

[4] Robert A. Freitas, Current status of nanomedicine and medical nanorobotics, J. Comput. Theor. Nanosci. 2 (1) (2005) 1–25.

[5] Robert A. Freitas Jr., Nanomedicine, Volume I: Basic Capabilities, Landes Bioscience, Georgetown, Texas, 1999.

[6] Erik Winfree, Furong Liu, Lisa A Wenzler, Nadrian C Seeman, Design and self-assembly of two-dimensional DNA crystals, Nature 394 (6693) (1998) 539.

[7] Paul W. K. Rothemund, Folding DNA to create nanoscale shapes and patterns, Nature 440 (2006) 297 EP, http://dx.doi.org/10.1038/nature04586, Article.

[8] Suping Li, Qiao Jiang, Shaoli Liu, Yinlong Zhang, Yanhua Tian, Chen Song, Jing Wang, Yiguo Zou, Gregory J. Anderson, Jing-Yan Han, Yung Chang, Yan Liu, Chen Zhang, Liang Chen, Guangbiao Zhou, Guangjun Nie, Hao Yan, Baoquan Ding, Yuliang Zhao, A dna nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo, Nature Biotechnol. 36 (2018) 258 EP, http://dx.doi.org/10.1038/nbt.4071.

[9] Phiset Sa-Ardyen, Alexander V. Vologodskii, Nadrian C. Seeman, The flexibility of dna double crossover molecules, Biophys. J. 84 (6) (2003) 3829–3837, http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1302964/. 15263[PII].

[10] Florian-Lennert Lau, Kristof Stahl, Stefan Fischer, Techniques for the generation of arbitrary three-dimensional shapes in tile-based self-assembly systems, in: VLIoT 2018 - International Workshop on Very Large Internet of Things (VLIoT 2018), Vol. 44, 2018, p. 9.

[11] T. Nakano, Molecular communication: A 10 year retrospective, IEEE Trans. Mol. Biol. Multi-Scale Commun. 3 (2) (2017) 71–78, http://dx.doi.org/10.1109/TMBMC.2017.2750148.

[12] Matthew J. Patitz, An introduction to tile-based self-assembly and a survey of recent results, Nat. Comput. 13 (2) (2014) 195–224, http://dx.doi.org/10.1007/s11047-013-9379-4.

[13] Alan M. Turing, Computing machinery and intelligence, in: Parsing the Turing Test, Springer, 2009, pp. 23–65.

[14] Florian Lau, Florian Büther, Bennet Gerlach, Computational requirements for nano-machines: there is limited space at the bottom, in: 4th ACM International Conference on Nanoscale Computing and Communication 2017 (ACM NanoCom'17), ACM, Washington DC, USA, 2017, pp. 11:1–11:6.

[15] James I. Lathrop, Jack H. Lutz, Scott M. Summers, Strict self-assembly of discrete sierpinski triangles, in: S. Barry Cooper, Benedikt Löwe, Andrea Sorbi (Eds.), Computation and Logic in the Real World: Third Conference on Computability in Europe, CiE 2007, Siena, Italy, June 18-23, 2007. Proceedings, Springer Berlin Heidelberg, 2007, pp. 455–464, http://dx.doi.org/10.1007/978-3-540-73001-9_47.

[16] Ho-Lin Chen, Ashish Goel, Error free self-assembly using error prone tiles, in: Proceedings of the 10th International Conference on DNA Computing, DNA'04, 2005.

[17] Erik Winfree, Renat Bekbolatov, Proofreading tile sets: Error correction for algorithmic self-assembly, in: J. Chen, J. Reif (Eds.), DNA Computing. DNA 2003, in: Lecture Notes in Computer Science, vol. 2943, 2004.

[18] Florian Büther, Florian Lau, Marc Stelzner, Sebastian Ebers, A formal definition for nanorobots and nanonetworks, in: The 17th International Conference on Next Generation Wired/Wireless Advanced Networks and Systems + the 10th Conference on Internet of Things and Smart Spaces (NEW2AN RuSMART 2017), Springer, St.Petersburg, Russia, 2017, http://dx.doi.org/10.1007/978-3-319-67380-6_20.

[19] Philipp Bende, Error Correction in Three-Dimensional Tile Assembly Systems (Master's thesis), Universität zu Lübeck, 2018.

[20] Matthew J. Patitz, Simulation of Self-Assembly in the Abstract Tile Assembly Model with ISU TAS. CoRR abs/1101.5151 (2009).

[21] Yaakov Benenson, Binyamin Gil, Uri Ben-Dor, Rivka Adar, Ehud Shapiro, An autonomous molecular computer for logical control of gene expression, Nature 429 (6990) (2004) 423–429.

[22] Ebbe S Andersen, Mingdong Dong, Morten M Nielsen, Kasper Jahn, Ramesh Subramani, Wael Mamdouh, Monika M Golas, Bjoern Sander, Holger Stark, Cristiano LP Oliveira, et al., Self-assembly of a nanoscale dna box with a controllable lid, Nature 459 (7243) (2009) 73–76.

[23] Ji Youn Lee, Soo-Yong Shin, Tai Hyun Park, Byoung-Tak Zhang, Solving traveling salesman problems with dna molecules encoding numerical values, Biosystems 78 (1) (2004) 39–47, http://dx.doi.org/10.1016/j.biosystems.2004.06.005.

[24] Qi Cheng, Gagan Aggarwal, Michael H. Goldwasser, Ming-Yang Kao, Robert T. Schweller, Pablo Moisset de Espanés, Complexities for generalized models of self-assembly, SIAM J. Comput. 34 (2005) 1493–1515.

[25] Paul W. K Rothemund, Nick Papadakis, Erik Winfree, Algorithmic self-assembly of dna sierpinski triangles, PLoS Biol. 2 (12) (2004) http://dx.doi.org/10.1371/journal.pbio.0020424.

[26] Erik Winfree, Renat Bekbolatov, Proofreading tile sets: Error correction for algorithmic self-assembly, in: DNA, in: Lecture Notes in Computer Science, vol. 2943, Springer, 2003, pp. 126–144.

[27] L. Felicetti, M. Femminella, G. Reali, P. Gresele, M. Malvestiti, J. N. Daigle, Modeling cd40-based molecular communications in blood vessels, IEEE Trans. Nanobiosci. 13 (3) (2014) 230–243, http://dx.doi.org/10.1109/TNB.2014.2340134.

[28] Ho-Lin Chen, Rebecca Schulman, Ashish Goel, Erik Winfree, Reducing facet nucleation during algorithmic self-assembly, Nano Lett. 7 (9) (2007) 2913–2919, http://dx.doi.org/10.1021/nl070793o.

[29] Nariman Farsad, H Birkan Yilmaz, Andrew Eckford, Chan-Byoung Chae, Weisi Guo, A comprehensive survey of recent advancements in molecular communication, IEEE Commun. Surv. Tutor. 18 (3) (2016) 1887–1919.

[30] Sasitharan Balasubramaniam, Pietro Lio', Multi-hop conjugation based bacteria nanonetworks, IEEE Trans. Nanobiosci. 12 (1) (2013) 47–59, http://dx.doi.org/10.1109/TNB.2013.2239657.

[31] Frank Walsh, Sasitharan Balasubramaniam, Reliability and delay analysis of multihop virus-based nanonetworks, IEEE Trans. Nanotechnol. 12 (5) (2013) 674–684, http://dx.doi.org/10.1109/TNANO.2013.2268389.

Florian Lau received his Bachelor in Computer Science in 2013 and his Master degree in Computer Science focussed on Computational Complexity Theory in 2015 respectively. Since 2016 he is a Ph.D. student at the University of Lübeck. He is a member of the Nano Group Lübeck, which was founded in 2016. His work is focused on DNA-based Self-Assembly Systems and their applications to Nanorobots and Nanonetworks.

Florian Büther received his diploma degree in Computer Science in 2010 at University of Lübeck. Since 2015, he works as a Ph.D. student at the Institute of Telematics at the University of Lübeck. His research topics include models for nanodevices and nanonetworks, routing algorithms for in-body medical nanonetworks and nanoscale computation and programming models.

Regine Geyer (geyer@itm.uni-luebeck.de) received her bachelor in Molecular Life Science in 2014 and her master's degree in Computer Science with the field of application in Bioinformatics in 2017 at the University of Lübeck, Germany. Since 2018 she is working as a research assistant at the Institute of Telematics, University of Lübeck, where she is currently pursuing a doctoral degree. She is a member of the Nano Group Lübeck founded in 2016. Her research focuses on the simulation of the working environment of medical nanobots and their communication networks with the goal to develop realistic medical application scenarios.

Stefan Fischer is a full professor in computer science at the University of Lübeck, Germany, and the director of the Institute of Telematics. He got his diploma degree in Information Systems and his doctoral degree in Computer Science from the University of Mannheim, Germany, in 1992 and 1996, respectively. After a post-doctoral year at the University of Montreal, Canada, he joined the newly founded International University in Germany, one of the first private universities in Germany, as an assistant professor in 1998. In 2001, he became an associate professor in Computer Science at the Technical University of Braunschweig, where he stayed until 2004, when he joined Lübeck University. His research interests are currently focused on software architectures for distributed systems and new networked systems technologies such as the Internet of Things and nano networks. He has (co-)authored about 200 peer-reviewed articles and seven books.