



Evolving cheating dna networks: a case study with the rock-paper-scissors game

Nathanaël Aubert, Quang Dinh, Masami Hagiya, Teruo Fujii, Hitoshi Iba,
Nicolas Bredeche, Yannick Rondelez

► To cite this version:

Nathanaël Aubert, Quang Dinh, Masami Hagiya, Teruo Fujii, Hitoshi Iba, et al.. Evolving cheating dna networks: a case study with the rock-paper-scissors game. European Conference on Artificial Life (ECAL-2013), 2013, Taormina, Italy. pp.1-8. hal-03175195

HAL Id: hal-03175195

<https://hal.archives-ouvertes.fr/hal-03175195>

Submitted on 19 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Evolving cheating DNA networks: a Case Study with the Rock–Paper–Scissors Game

Nathanaël Aubert¹, Quang Huy Dinh¹, Masami Hagiya¹, Teruo Fujii³,
Hitoshi Iba², Nicolas Bredeche^{4,5}, Yannick Rondelez³

¹ Graduate School of Information Science and Technology, The University of Tokyo, Japan

² Graduate School of Engineering, The University of Tokyo, Japan

³ LIMMS/CNRS-IIS, Institute of Industrial Science, The University of Tokyo, Japan

⁴ UPMC Univ Paris 06, UMR 7222, ISIR, F-75005, Paris, France ⁵ CNRS, UMR 7222, ISIR, F-75005, Paris, France

naubert@is.s.u-tokyo.ac.jp

Abstract

In models of games, the indirect interactions between players, such as body language or knowledge about the other's playstyle, are often omitted. They are, however, a rich source of information in real life, and increase the complexity of possible strategies. In the game of rock-paper-scissors, the simple monitoring of the opponent's move *before* it was played is a sufficient condition to trigger an arms race of detection and misinformation among evolved individuals. The most interesting aspect of those results is that they were obtained by evolving purely chemical reaction networks thanks to an adapted version of the famous NEAT algorithm. More specifically, those individuals were represented as biochemical systems built on the DNA toolbox, a paradigm that allows both easy *in-vitro* implementation and predictive *in-silico* simulation. This guarantees that the specific motives that emerged in this competition would behave identically in a test tube, and thus can be used in a more generic context than the current game.

Introduction

The game of rock-paper-scissors, while being simple, can actually lead to interesting dynamics when it is played multiple times in a row. In particular, each player will try to “read” their opponents in the hope of getting the upper hand. However, if psychological factors are not taken into account, that is, if players are purely logical, game theory predicts that after a while, the optimal strategy becomes to play randomly with no bias among the three possible moves (Smith, 1993). Variations of the basic rules exist, but are expected to display the same kind of behaviors (from the point of view of game theory) as the classic three moves.

Interestingly, this game can be a good description of many mechanisms ranging from reproductive strategies of some species of lizards (Sinervo and Lively, 1996) or bacteria (Kerr et al., 2002) to oscillations in a gene regulatory circuit (Elowitz and Leibler, 2000). In all cases, there are three possible moves, each strong against another and weak against the remaining one. This usually leads to dynamical behaviors where the different players are constantly invading each other, forming complex spiral structures in two dimensional systems (Kerr et al., 2002; Reichenbach et al., 2007). Even

real life examples, such as the lizard example, display oscillations in population size, with a turnover of approximately six years, based on the field data of (Sinervo and Lively, 1996). Those dynamics may degenerate into a uniform population depending on the initial conditions, or such parameters as the mobility of the players. On the other hand, they may also occur even in a well-mixed system, where there is no spatial compartmentalization to protect diversity, if a given move gets stronger when it is less frequent (Freen and Abraham, 2001) or if the system never stalls, like in the repressilator (Elowitz and Leibler, 2000).

However, all those examples either suppose or require that a given individual will always “play” the same move. Indeed, the lizard will always have the same size and coloration, bacteria the same genotype and genes in the repressilator are not expected to arbitrarily change which target genes they inhibit. From a strategic point of view, more possibilities open when each agent can decide, at each time, which move he wants to put forward. In such a case, some form of knowledge of the opponent becomes necessary in order to infer his probable next move and play accordingly. This knowledge is obtained from two sources: cheating and analysis of the opponent previous moves. “Cheating” here designates the fact of obtaining clues about an opponent from its behavior just prior to the game, not in the negative sense of making a game uninteresting by bypassing the rules. Note that cheating in this sense is both an integral part of most human plays and of biological strategies, and in any way is an essential ingredient of any physically instantiated game. In fact, instantaneous moves and decisions are not possible in a physical world, which means that information is always leaked somehow. This fact was used by the Ishikawa laboratory in Japan to program a robot hand (Namiki et al., 2003) reacting fast enough to hand gestures to be able to always win against a human (video online).

While both cheating and strategic analysis requires significant abilities and are generally associated with intelligent players (or at least, players with intents), we wanted to demonstrate in this paper that purely molecular systems are also capable of intricate strategies, whose complexity can

be comparable to that of real players. Indeed it has been recently demonstrated that Turing universality can be achieved through the sole use of chemical reactions (Magnasco, 1997; Soloveichik et al., 2010; Cardelli, 2011). Moreover, practical bottom-up approaches have been proposed to actually instantiate arbitrary reaction networks (Seelig et al., 2006; Qian and Winfree, 2011). However, experimentally, only relatively simple tasks (equivalent in complexity to those performed by the most basic electronic circuits) have been demonstrated. Even from a theoretical standpoint only quite simple systems have been proposed, very far from the intricacy observed in the case of cellular regulation maps, or even bacterial behaviors.

The individuals we evolved were defined as entities from the DNA toolbox (Montagne et al., 2011), a particular paradigm to define DNA-based computing systems. In particular, we build on a unique feature of the DNA toolbox, which is to couple a generalized experimental strategy for the *in vitro* building of reaction networks to the availability of straightforward (if large, from the point of view of equation solving) quantitative models. These models allows exact mathematical predictions and thus allow to perform both *in vitro* and *in silico* designs in parallel.

Individuals were evolved through an adapted version of NeuroEvolution of Augmenting Topologies (NEAT) (Stanley and Miikkulainen, 2002), dubbed bioNEAT, using a fitness function based on how well they fared in a population-wide tournament. To our surprise, the apparition of a basic memory was not hard, but was almost immediately discarded, as it was not able to compete against cheating. Due to the necessity of having both players in the same well-mixed environment, it was much more efficient for an individual to actually develop a way to monitor the actions of its opponent while hiding its own move. When pushed to the extreme, this strategy produced interesting dynamics where individuals went through multiple moves before the end of the countdown, trying to settle into a winning position, eventually leading to some fashion of oscillatory systems. The mechanisms used for those purpose were interesting in themselves, including concentration comparators or system with multiple levels of activation, giving, through motif mining, insight into the possibilities of the DNA-toolbox. This showed that indeed, the behavior of purely molecular systems, corresponding to a realistic, directly implementable chemistry, can be interpreted in terms of complex strategic planning.

Related Work and Current Contributions

Our work builds on multiple sources since it mixes design by genetic algorithm with molecular programming. Game theory was also an important source of inspiration, and was useful to check that our evolved individuals are playing in a way that differs from hypothetical “perfect” players.

Rock-paper-scissors

There are also many previous works related to the game of rock-paper-scissors. However, to the best of our knowledge, they either use individuals which are only capable of playing one move, or link existing dynamics to an instance of the game. The evolution game theory study in (Smith, 1993) is the closest to our work, but lacks the added dimension that comes with dealing with cheating or leak of information (Cook et al., 2012). While DNA-based systems can hardly be described as having any form of intelligence, it is easy to rationalize their behavior as cheating, a very real possibilities among human players that is not taken into account in (Smith, 1993).

Motif Mining

The idea of using DNA computing to play games has been previously introduced (Macdonald et al., 2008). Finding systems able to play a game is in itself a challenge that leads to developing new structures, and potentially solve issues related to real life problems. However, the use of evolutionary algorithms (Eiben and Smith, 2003) stand as a promising candidate to search for interesting reaction circuits. From the structural point of view, the analysis of the fittest individuals of specific runs revealed common functional motifs, which may help build new systems. This is the fundamental approach of synthetic biology, in which biological modules are recombined to perform engineered operations (Purnick and Weiss, 2009). In particular, it was interesting to note that, although actual patterns may vary from individuals to individuals, it was possible to classify them into rough generic categories. This could be used to create minimal libraries of structures for dynamic systems, that is, off-the-shelves building blocks like those defined in (Rodrigo et al., 2011). Such libraries would in turn allow the fast and reliable development of complex DNA-based systems. While, in our case, the structures evolved by the algorithm are possibly not generic enough to be useful in any given context, they still have potential applications for the design of a variety of such systems.

Model

The DNA toolbox

The DNA toolbox (Montagne et al., 2011; Padirac et al., 2012) is a set of three modules designed to reproduce gene regulation networks dynamics with a simple framework. Those modules, namely activation, autocatalysis and inhibition, use solely DNA strands and enzymes, making both modelization and implementation of systems straightforward (at least when compared to the *in-vivo* lego networks of synthetic biology). DNA sequences have two possible roles: either signal (simply designated as sequences in the following) or templates. The templates are the backbone of DNA toolbox systems, and are used to generate a specific signal

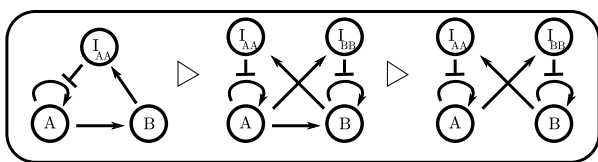


Figure 1: Graphical representation of systems from the DNA toolbox. Nodes represent sequences while arrows represent templates. The Oligator (left) can be mutated into a bistable in two steps. First, an autocatalysis connection B to B with an inhibition from A is added. Then, the activation from A to B is removed. Note that those two operations may happen in any order.

from another signal. Specific sequences can also be generated to inhibit a given template. Since they represent the “code”, templates are kept stable over time, and are chemically protected against enzymatic activity that could affect them. Signal sequences, on the other hand, are continuously degraded to keep the system dynamic.

The important feature of the DNA toolbox activatory and inhibitory modules is that they are arbitrarily connectable to each other. The designer of the network freely defines the pattern of interactions by assigning the sequences of the template through Watson-Crick complementarity. For example, a cascade of activation reaction is obtained by mixing a number of bidomain templates such as AB , BC or CD , where A , B , C , D , and so on represent orthogonal 11mers. The Oligator from (Montagne et al., 2011), a simple oscillator, is obtained by combining the three templates AA , AB and $Blaa$ (where Iaa represent the inhibitor of AA). The graph of this system can be seen in Figure 1, left.

One interest of the toolbox in the scope of genetic algorithms is that any modification of the “genome” of an individual (that is, the sequences and templates it is made of, not to be confused with the hypothetical genome their actual DNA strings are encoding) still yields a valid individual (albeit a possibly uninteresting one), and that a wide range of possible behaviors are very few modifications apart. For instance, bioNEAT (see next Section) can jump in two steps from the Oligator (Montagne et al., 2011) to Padirac *et al.*’s bistable system (Padirac et al., 2012), as shown in Figure 1. This helps the algorithm navigating the search space more efficiently, as well as preventing, to some degree, the trap of local optima.

Individuals and encoding The individuals we consider are chemical reaction networks playing rock-paper-scissors. Each possible move (rock, paper or scissors) is mapped to a specific chemical species (DNA sequences, more specifically signal sequences from the DNA toolbox). Those species are fixed in advance, so that they are always present. Individuals also have references linking to potential oppo-

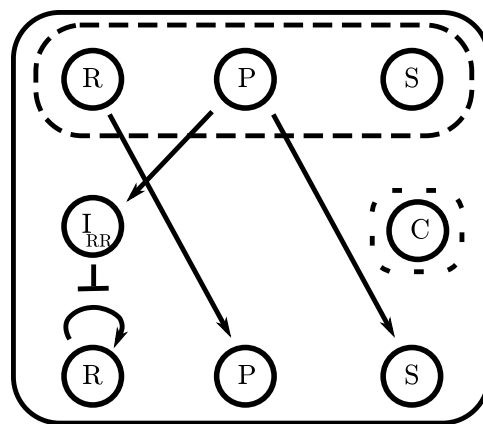


Figure 2: Simple cheating individual displaying both direct and indirect monitoring. Nodes in the dashed box are references to the opponent’s sequence (up) or to the clock (right). By default, this individual will play rock (R). If its opponent plays rock or paper (P), it will update to play the winning move. Note that this individual does not use the clock.

nents’ corresponding sequences. The main goal of this interface is to allow individuals to react to the opponent’s moves and adapt their strategy over time. Finally, all individuals have a reference to a common clock species, giving them a sense of time. An example of individual is shown in Figure 2.

Individuals are pitted against each other in matches made of ten rounds. The beginning of a round is marked by a spike from the clock sequence. At the end of a round, roughly 20 times the clock’s half-life later, an individual’s move is decided by which of its move sequences has the highest concentration. If the two highest or all such concentration are not different by at least a given threshold, the move is considered invalid, granting the victory to the opponent. Individuals can potentially memorize their opponent’s strategy, since there is no reset between rounds.

Simulations The simulation itself was kept simple, with a model similar to that of Padirac *et al.*. In particular, this model doesn’t take into account enzyme saturation. This prevents some advanced strategies (since saturating enzymes may be in itself a way to kill one’s opponent, thus winning by default) and allows individuals to grow without limitations, continuously increasing their size. Since enzymatic saturation creates hidden couplings between the nodes (Rondelez, 2012), removing it was taken as a step to insure the readability of the results. Thanks to this, the behavior of the network - and hence the individual’s strategy - is directly encoded by the networks of cross regulations between the nodes, and not by various type of competitive inhibitions acting at a global level. Using this simplified model is also a compromise between computational requirements and pre-

cision, but any observed behavior should be obtainable in real *in-vitro* experiments.

bioNEAT: NEAT for Reaction Networks

The evolution of individuals was done by using a modified version of NeuroEvolution of Augmenting Topologies (NEAT) (Stanley and Miikkulainen, 2002), adapted to perform with simulated individual networks built using the DNA toolbox paradigm instead of artificial neural networks. The evolution itself was performed through multiple runs and tweaking of the fitness function.

NEAT

NEAT is a state-of-the-art evolutionary algorithm designed to evolve both the topology and the parameters of neural networks, while keeping them as simple as possible. This is done by starting from very simple individuals, and progressively complexifying them in a competitive process. This is performed through the addition of new nodes and connections, while at the same time modifying the weight of existing ones.

The major strength of NEAT is that it keeps tracks of when specific connections or node were added in the ancestry line. This allows to perform meaningful cross-over: identical elements present in two individuals, are automatically recognized and matched during the creation of a new individual from two parents. Additionally, mismatching elements from the fittest individual are also passed along.

NEAT also performs speciation to protect innovation that could require more than one step to find a new, better solution to the problem at hand. Specifically, the size of a species depends on the average fitness of its individuals, preventing one type of solution to completely invade the population. Moreover, speciation is easily performed, since the history of evolution of individuals is saved, giving a straightforward distance between individuals based on the genes they possess.

bioNEAT

Due to the initial resemblance between reaction network and artificial neural network, NEAT stands as a relevant option for optimizing toolbox-based systems. In particular, systems from the DNA toolbox have a straightforward edge/node graph representation similar to neural networks: DNA sequences can be directly mapped to nodes, and connections with positive weights are equivalent to activation links. However, the DNA toolbox cannot be directly implemented using the original NEAT for two reasons. Firstly, additional parameters regarding sequences stability and initial concentration must be added. Secondly, negative links targetting nodes must be replace by inhibitory links targetting arcs. To address these issues, we introduce bioNEAT, a NEAT-derivative that is able to optimize reaction networks.

A first feature of bioNEAT is to allow the GA to not only modify the “weight” of connections (that is, the concentration of DNA template, in our representation), but also the relevant biological parameters (such as the thermodynamical stability of DNA sequences and their initial concentrations). The thermodynamical parameters of the move sequences was fixed to prevent individuals to use extremely stable sequences to saturate the monitoring of their opponents. In the particular case of the experiments described hereafter, we also prevented activations toward the opponent or the clock.

The second feature of bioNEAT addresses the asymmetry between activation and inhibition process that is inherent to the DNA toolbox, and which cannot be modelled as a classic neural networks link with positive and negative weights. While the sign of a neural weight simply encodes the type of the connection and target a node, a DNA toolbox’ inhibitor targets an *edge* (and impact only one of the output from the source node) rather than a node. Moreover, an inhibitor cannot be instantiated without the template it inhibits. As a consequence, bioNEAT protects the addition of an inhibitory connection (and removal of a particular template) during evolution. Then, bioNEAT produces reaction network with inhibitory connections from node to link.

Fitness Score

Scoring of an individual uses a lexicographic fitness function taking place in two steps. First, the individual has to beat the three most basic possible players, playing respectively only rock, paper or scissors. This ensures that our individuals are able to play all moves, and to play them discerningly. Individuals unable to pass this test are awarded a very small fitness, based on the number of rounds they have won, directing the evolution toward basic strategies. On the other hand, individuals which were able to pass the test are awarded the right to enter the second phase.

The second phase is a simple tournament among all remaining individuals: each of them has to fight each of the others. The fitness is then based on the amount of correct moves made in total. A sample match is shown in Figure 3. Because of this, the evolutionary pressure forces the individuals into an arms race, to be able to defeat as many opponents as possible.

Results

Results were obtained by evolving individuals in 10 separate runs, always starting from a uniform population of individuals with autocatalysis on the rock sequence (thus playing always rock). A typical run involved 200 generations of a population of 100 individuals. bioNEAT speciation control loop is adjusted to keep the number of species as close as possible to 10. Other relevant parameters are shown in Table 1. Over the course of the experiment, various kind of strategies emerged before getting outdated or integrated into

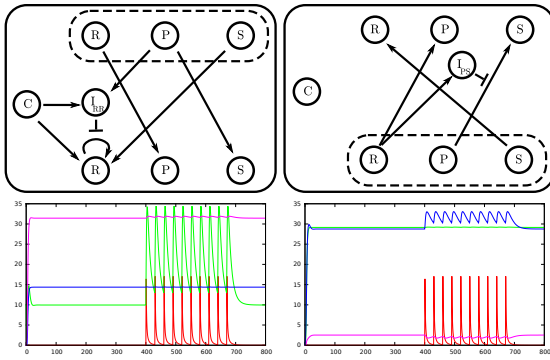


Figure 3: Two fighting individuals. References to the opponent’s nodes are shown in the dashed box. Top: the actual network of those individuals. Bottom: the corresponding behavior over time. The color code for sequences concentration is red for the clock, green for rock, blue for paper and purple for scissors. The individual on the right has a better comparison mechanism than the individual on the left, as shown by the fact that it has the correct move before the match starts. However, the individual on the left uses the clock to fake switching his move from scissors to rock, which coerce its opponent to update its move to paper. Just before the round is validated, the individual on the left changes its move again to scissors, winning each hands.

more complex control systems. However, in our runs, a stable group of species typically appeared after 50 to 100 generations and quickly took over the population until the end of the run. They represent individuals which had developed part or all of the mechanisms explained later in this Section, and the apparent stability was only due to a constant arms race, where individuals kept adding more and more modules, while those who couldn’t keep up were discarded. However, since our fitness can only compare individuals among a given generation, its evolution over time does not reflect the global improvement of individuals. This prompted us to perform a post-mortem analysis of our individuals by making the best of each generations of a given run fight each other, highlighting a progressive improvement of our individuals, as shown in Figure 4. In particular, the logarithmic shape of the curve goes well with the idea that the efforts required to overcome one’s opponents are greater and greater as the simplest strategies get commonly countered.

Cheating

The easiest, and thus first strategy evolved is actual cheating. Since they have references to what each other will play, and continuous access to current concentrations, the individuals monitor the action of their opponent and try to play accordingly. A minimal example is shown on Figure 2. Cheating can be of two kinds: either using a direct connection (“if my opponent plays rock, I will play paper”), or an inhibition (“if

General parameters

Population size 100
 Number of generations 200

Speciation parameters

Targeted number of species 10
 NEAT compatibility parameters $c_1 = c_2 = 1; c_3 = 0$
 Initial speciation threshold 0.6
 Minimal threshold 0.1
 Threshold update ϵ 0.03

Mutation parameters

P(Mutation only) 0.25
 P(Parameter mutation) 0.9
 Otherwise P(Add node) 0.2
 Otherwise P(Add activation) 0.2
 Else add inhibition
 P(Connection disabling) 0.1
 P(Gene mutation (for each node)) 0.8

Crossover parameters

P(Interspecies crossover) 0.01
 P(Re-enabling gene) 0.25

Table 1: Parameters used to evolve individuals

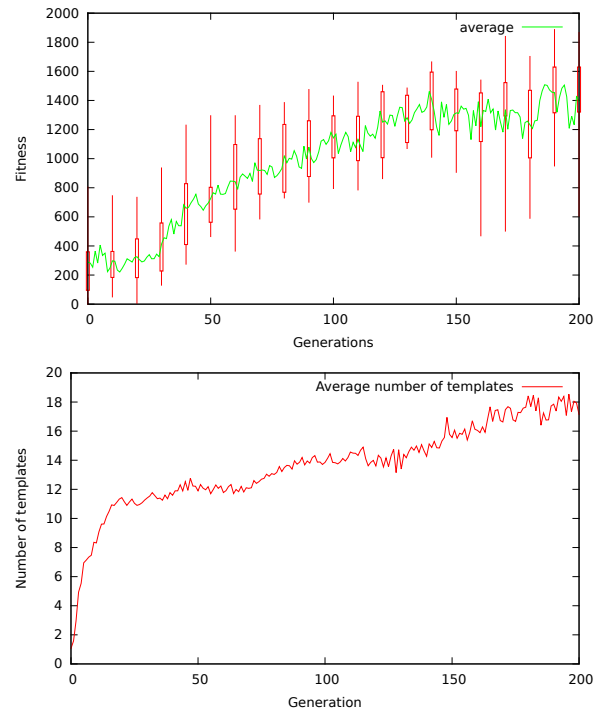


Figure 4: Top: average *a posteriori* fitness of the best individuals, as well as minimum, maximum, first and third quartiles. While noisy, the curve still shows an increasing trend similar to that of a logarithm. Bottom: the average number of templates in individuals over generations in a typical run. The trend is similar to that of the fitness, showing that bloating stays within acceptable limits.

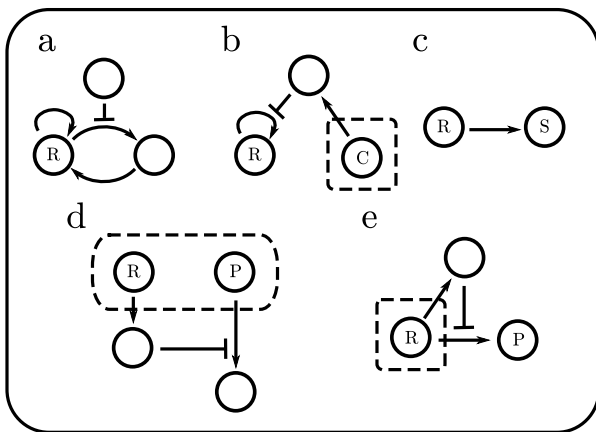


Figure 5: Basic mechanisms observed in individuals. (a.) Noise generation with two activation level. When the additional path is inhibited, the main sequence will still have a high concentration, but not high enough to be this turn's move. (b.) A given move's concentration is kept low for some time by being inhibited by the clock sequence *C*. (c.) A very simple feint: while pretending to play rock (the sequence *R*) has a non-zero concentration), the individual is actually playing scissors (*S*), which would win against the expected reaction of the opponent. This mechanism is often decorated with various other systems to balance the concentrations of one sequence relatively to the other. (d.) Simple comparison mechanism. The reaction path from the opponent's move will only be activated if the concentration of paper (*P*) is high enough, compared to the concentration of rock (*R*). (e.) A fold change detector, allowing the monitoring of the increase in the concentration of the rock (*R*) sequence of the opponent. Often, the detection will happen after a first amplification of the monitored signal.

my opponent plays rock, I will not play scissors"). Cheating leads in some cases to the apparition of oscillatory behaviors, as both individuals are both trying to play the winning move.

Defense mechanisms

Once cheating appears, it quickly spreads among the whole population, either by cross-over, elimination of individuals which could not adapt, or by parallel discovery of the mechanism. From there on, the only way to improve is to develop mechanisms against the other cheater's spying while at the same time improving the monitoring of its current move. Many defenses were expressed among the evolved individuals, but can mainly be separated into five categories: noise generators, stealth, feint, concentration comparators and fold change detectors. Representatives of all those categories are shown in Figure 5.

Noise generators are the easiest form of defense. Since it is fair to assume that the opponent will monitor at least

two move sequences to decide its own next move, a simple yet efficient way to keep it off track is to continuously generate all sequences. This is a valid action, since only the highest sequence decides which move is played. Having a weak autocatalytic connection is enough, as long as there is a way for the other sequences to become lower (remember that an individual has to be able to play all moves to have a good fitness). Often, such sequence will have an additional catalytic loop using an additional sequence. This loop is only activated when this sequences is supposed to be played. This simple mechanism allows the individual to have multiple activation levels (by opposition to just "on" and "off"), with a better control on the final concentration of the target sequence rather than using activation mechanisms from different possibly not trustworthy part of the system.

Stealth is the complementary of noise generation. Instead of hiding one's true move among decoys, it is kept at a concentration as near to zero as possible until the last moment. This technique relies on monitoring the clock sequence, since timing is extremely important. The clock sequence is used to generate a large amount of timer, which in turn inhibits a specific move. If the inhibition is stable enough, the target sequence will be kept low until the timer has been degraded. If the delay is not long enough, the opponent will still have time to read and adapt. On the other hand, if the delay is too long, the move will not be valid. Part of the system dedicated to this mechanism seems to be very stable over generations, since it is based on a delicate balancing of parameters where any change can prove deadly.

Feint resembles closely the previous two strategies, but uses a different structure. In this case, the individual spoofs a specific move (say "rock"), but this very move also activates the generation of the real move (for instance "scissors"), often through a long activation path to generate delay. It relies on the fact that the opponent will try to adapt to the perceived move, and won't be able to react in time to the change. The system may be reset by the clock, or by a change in the opponent's perceived move.

As the direct monitoring of sequences became less and less reliable, structures to compare absolute concentrations as well as detect sudden modifications became more and more common. Concentration comparison is done through the inhibition of a reaction path if its activation is not strong enough compared to the reference. Since this inhibition originates from the monitoring of another sequence, the first pathway is activated only if the first sequence has a higher concentration. Of course, by tuning the strength of pathways and inhibition, it is possible to have more specific control over the targeted ratio between the two sequences. For instance, it would be possible to slightly modify the system to inhibit the reaction path only if the compared sequence has a concentration multiple times higher than the reference sequence. This defense mechanism is used to counter noise generators and feints.

The last technique commonly spread among individuals is a way to detect concentration increase. While concentration comparison is able to detect that a stealthy move is being played, it is only able to do so once the move became dominant (which, if the other player is timing right, should be too late). However, by using a monitoring coupled with incoherent feedforward, individuals are capable of detecting rapid variations in concentration, which would be a sign that their opponent is about to switch their move. Some individuals also *pretended* to switch their move to throw such defense technique off guard, but this was quickly countered by a mix of both direct comparison and incoherent feedforward.

Memory vs cheating

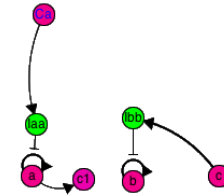
Quite early on, individuals with a basic memory, such as the bistable from Figure 1, appear in the population. However, those individuals were too “naive” in the sense that they had no defense against cheaters. Moreover, cheating requires about the same amount of mutations to appear, or even less if partial (that is, the individual can read *some* moves, but not all). For this reason, it seems that it is much more advantageous for individuals to focus only on attack and defense. This prevented the reappearance of memory in later generation, leading to purely reactive individuals.

The arms race

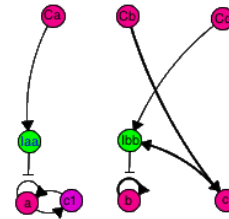
Looking at individuals over time shows the apparitions of the different cheating and defense mechanisms over time, with a noticeable complexification of the best individuals. Figure 6 shows such individuals at different times of a specific run, highlighting the apparition of various mechanisms.

The logical conclusion of this evolution strategy is that individuals with high fitness in a given generation have very little, or even no structures that are not related to cheating and defeating. Even when they exist, such structures are mutated during the next few generations to serve some attack or defense purpose. We performed an *a posteriori* evaluation of the fitness to check whether this increase in individuals size was indeed justified or only bloating. By performing this evaluation, we get a sense of the improvement of individuals over time that cannot be deduced from the lexicographic fitness used for evolution, since the later one only compares individuals from a given generation. The fitness itself is computed by making the best individual of all generations fight each other and score points in the same fashion than in the second part of the lexicographic fitness.

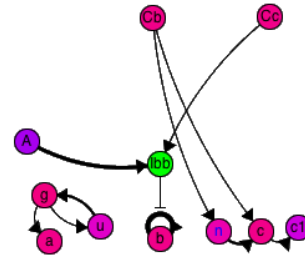
The trend of the *a posteriori* fitness also implies that there is no cyclic effect. While the lexicographic fitness guarantees that all individuals have the capacity of playing any move given the right conditions, there could be more advanced strategy displaying such cyclic dynamics. For instance, individuals using stealth are beaten by individuals using incoherent feedforwards, which could have been, in



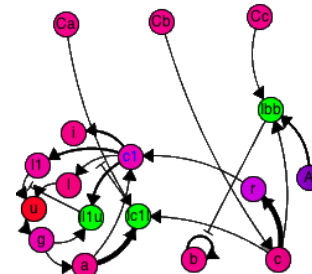
Generation 10: partial cheating.



Generation 14: complete cheater.



Generation 109: stealth. The clock sequence (here designated A) hides a move (b).



Generation 122: fold change detector. The sequence *c* both activates and inhibits the creation of *a*. However, the activation path is longer than the inhibition path, meaning that *a* (rock) is only activated by this module if the concentration of *c* (scissors) is decreasing. Since *c* is directly linked to the opponent’s *b* (paper), this individual is protected against stealthy play of *b*.

Figure 6: Individuals generated during a run. The color of activation nodes indicates their stability, going from red (very unstable) to blue (very stable). Green nodes are inhibitors. The notation for the moves rock, paper and scissors is respectively *a*, *b* and *c*. References to the opponent’s sequences are designated by a leading *C*. *A* represents the clock.

turn, beaten by another strategy that is weak against stealth. Since the fitness increase is monotonic (if we ignore the noise), we can conclude that the arms race is open-ended, with complexification of individuals the only possible way to improve.

We could also note that the arms race pushes individuals to perform well within their own ecosystem, but not always optimally. For instance, the individual from generation 122 in Figure 6 only defends against stealthy changes in the concentration of “paper”, leaving it open to the exact same strategy, if performed on another move. However, it is easy for a human designer to take inspiration from those modules to create an “optimal” player.

Conclusion

In this work, our first hope was to observe the emergence of memory to allow non-trivial strategies at rock-paper-scissor using bioNEAT, a modified version of NEAT designed to evolve chemical reaction networks from the DNA toolbox. However, the very rules, derived from experimental settings, we set for the games prevented this mechanism from being efficient. Instead, increasingly complex cheating seemed to be the best answer. However, this is not the only thing we learned from this exercise. While having DNA systems compete against each other and evolve new (cheating) strategies can be a goal in itself, the systems evolved along the way gave us also more insight about DNA computing systems. In particular, it was possible to observe the emergence of particular structures with interesting dynamics, which may prove useful to a human trying to develop DNA systems, like with the libraries of (Rodrigo et al., 2011). It could be also interesting to make individuals compete against a human designed “optimal” cheater and see if they can evolve even more advanced strategies to counter it. Furthermore, since the DNA toolbox mimic the behavior of gene regulatory circuits (Montagne et al., 2011), an open question would be whether those mechanisms appear in real life or if they are only valid in the toolbox. Also, it would be interesting to extend the current systems to take into account reaction-diffusion and be able to play more complex games. There is little doubt that such systems will have their own share of remarkable mechanisms.

References

Cardelli, L. (2011). Strand algebras for dna computing. *Natural Computing*, 10(1):407–428.

Cook, R., Bird, G., Lünser, G., Huck, S., and Heyes, C. (2012). Automatic imitation in a strategic context: players of rock–paper–scissors imitate opponents’ gestures. *Proceedings of the Royal Society B: Biological Sciences*, 279(1729):780–786.

Eiben, A. E. and Smith, J. E. (2003). *Introduction to Evolutionary Computing*. Springer-Verlag.

Elowitz, M. B. and Leibler, S. (2000). A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338.

Frean, M. and Abraham, E. R. (2001). Rock–scissors–paper and the survival of the weakest. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 268(1474):1323–1327.

Kerr, B., Riley, M. A., Feldman, M. W., and Bohannan, B. J. (2002). Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors. *Nature*, 418(6894):171–174.

Macdonald, J., Stefanovic, D., and Stojanovic, M. N. (2008). Dna computers for work and play. *Scientific American*, 299(5):84–91.

Magnasco, M. O. (1997). Chemical kinetics is turing universal. *Physical Review Letters*, 78(6):1190–1193.

Montagne, K., Plasson, R., Sakai, Y., Fujii, T., and Rondelez, Y. (2011). Programming an in vitro dna oscillator using a molecular networking strategy. *Molecular systems biology*, 7(1).

Namiki, A., Imai, Y., Ishikawa, M., and Kaneko, M. (2003). Development of a high-speed multifingered hand system and its application to catching. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 3, pages 2666–2671. IEEE.

Padirac, A., Fujii, T., and Y., R. (2012). Bottom-up construction of in vitro switchable memories. *Proceedings of the National Academy of Sciences*, 109(47):E3212–E3220.

Purnick, P. E. and Weiss, R. (2009). The second wave of synthetic biology: from modules to systems. *Nature Reviews Molecular Cell Biology*, 10(6):410–422.

Qian, L. and Winfree, E. (2011). Scaling up digital circuit computation with dna strand displacement cascades. *Science*, 332(6034):1196–1201.

Reichenbach, T., Mobilia, M., and Frey, E. (2007). Mobility promotes and jeopardizes biodiversity in rock–paper–scissors games. *Nature*, 448(7157):1046–1049.

Rodrigo, G., Carrera, J., and Jaramillo, A. (2011). Computational design of synthetic regulatory networks from a genetic library to characterize the designability of dynamical behaviors. *Nucleic acids research*, 39(20):e138–e138.

Rondelez, Y. (2012). Competition for catalytic resources alters biological network dynamics. *Physical Review Letters*, 108(1):018102.

Seelig, G., Soloveichik, D., Zhang, D. Y., and Winfree, E. (2006). Enzyme-free nucleic acid logic circuits. *science*, 314(5805):1585–1588.

Sinervo, B. and Lively, C. M. (1996). The rock–paper–scissors game and the evolution of alternative male strategies. *Nature*, 380(6571):240–243.

Smith, J. M. (1993). *Evolution and the Theory of Games*. Springer US.

Soloveichik, D., Seelig, G., and Winfree, E. (2010). Dna as a universal substrate for chemical kinetics. *Proceedings of the National Academy of Sciences*, 107(12):5393–5398.

Stanley, K. O. and Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99–127.