

# A Representation of Artificial Spin Ice for Evolutionary Search

Arthur Penty and Gunnar Tufte

Department of Computer Science,  
Norwegian University of Science and Technology, Trondheim, Norway  
arthur.penty@ntnu.no

## Abstract

Arrangements of nanomagnets known as artificial spin ices show great potential for use in unconventional computation. The majority of exploratory work done in this area considers just a small handful of well studied geometries (nanomagnetic arrangements), and uses them as if they were a black box. Here we detail a novel representation of artificial spin ice geometries, which lends itself to the tuning and evolutionary search of geometries. Using our representation we present geometries tuned to exhibit a desired computational or meta-material property. This is the first example of such a search performed on artificial spin ice.

## Introduction

New materials and novel computational paradigms are of great interest, as traditional silicon based computing systems struggle with scaling issues due to the slow-down of Moore's law (Moore, 1965), the power wall (Bose, 2011), and architectural challenges (Dennard et al., 1974). The architectural challenges do not relate to the silicon substrate but originated in the Turing-Von Neumann (Turing, 1937; von Neumann, 1945) concept of global control and continuous data and instruction movement.

In contrast to the global sequential principles of conventional computers, unconventional computation (Teuscher and Adamatzky, 2005), and material computation (Stepney, 2008) may exploit vast parallelism and emergent phenomena, e.g., self-organisation, in combination with bio-inspired design methods to achieve a physical system capable of extremely effective computation.

Artificial spin ice (ASI) has become a medium of interest for both its behaviour as a meta-material (Stepney, 2008), and its computational potential (Jensen et al., 2018). ASI is a ferromagnetic meta-material consisting of a large array of nanomagnets. In most of work on ASI, the nanomagnets are organised in fixed repeated patterns termed geometries. Figs. 1a to 1c illustrate three such patterns: Square, Pinwheel and Kagome. The nanomagnets interact locally providing a material with complex emergent behaviour (Skleinar et al., 2019). ASI's self-organisation properties, together

with their rich dynamics based on a large number of non-linear elements, make it an intriguing substrate for material computation.

ASI arrays are easy to scale, i.e., scaling the number of elements from a few to millions. Furthermore, ASI systems can be perturbed externally, e.g., using global or local magnetic fields as to influence the dynamics.

ASI is a substrate well suited to neuromorphic computing (Monroe, 2014) and evolution in materio (Miller et al., 2014). The external magnetic field and the geometry are the main adjustable properties that influence the dynamical and emergent properties of an ASI. In recent work by Jensen and Tufte (2020) and Hon et al. (2021) an external magnetic field was used to tune the ASI dynamics toward computational properties. Here we take the less explored route of searching for novel ASI geometries that provide the sought after properties.

Evolutionary Algorithms (EAs) are a good choice for exploring new substrates, particularly when the relationship between the configuration of the system and the emergent behaviour is mostly unknown. However, the number of elements in ASIs can be a challenge for EAs, if using a representation requiring each of many magnet to be parameterised independently. To support the the possibility of exploring the scalability of ASI we propose using a genotype-phenotype mapping inspired by tiling (Downing, 2005) where the representation could be used to generate geometries at different sizes.

We put forward a new representation for ASIs, to facilitate the evolutionary search of their geometry. We evaluate our representation and the geometries it produces, using a material property and a computational property, assessing resilience and diversity respectively. The two quite opposite properties are chosen to show that ASI geometries are truly evolvable, it is possible, using our representation, to move smoothly an incrementally through the search space. Further, the scalability of the representation is explored by investigating whether the novel evolved ASI geometries produced can be scaled up in size whilst a dynamic property is maintained. The tiling approach offers the possibility to

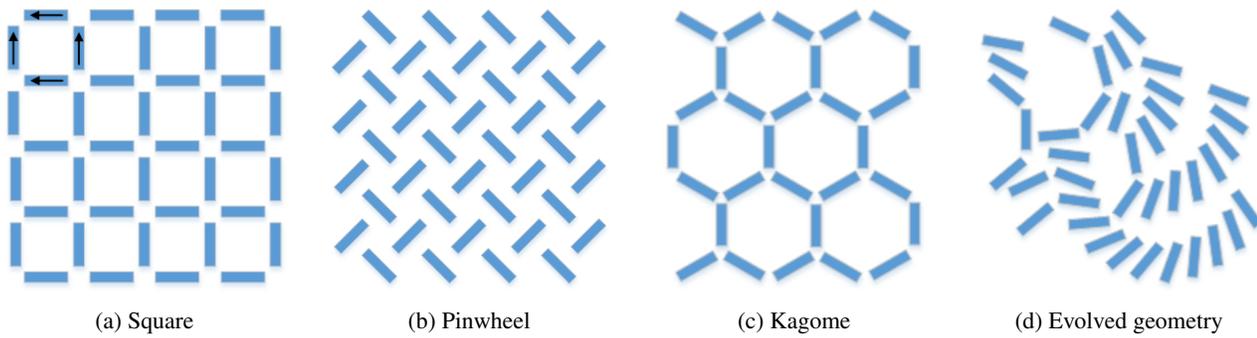


Figure 1: Example of standard ASI geometries (a), (b) and (c). (d) illustrates an instance of an evolved ASI geometry.

keep the search space (genotype space) small or constant, whilst the size of the ASI (phenotype) can be specified independently as part of the mapping process. The evolved ASIs are evaluated toward dynamic behaviour, i.e., the fitness is based on trajectories through the state space.

All ASI systems are simulated using the flatspin simulator (Jensen et al., 2020).

### Artificial spin ice

Originally, ASIs were developed as nanosystem substrates for studies into geometrical frustration (Wang et al., 2006). In later years the term ASI has become broadened to include any 2D arrangement of nanomagnets. The vast majority of ASI experiments deal with uniform pattern of nanomagnets, e.g., Square (Wang et al., 2006) or Pinwheel (Macêdo et al., 2018) (Figs. 1a and 1b), due to the need for somewhat predictable behaviour in physical experimental samples. However, there is no requirement for an ASI to be restricted to repeating patterns, as long as the design constraints, e.g., minimum distance between any two magnets, are fulfilled. Fig. 1d illustrates an example of a novel evolved ASI geometry.

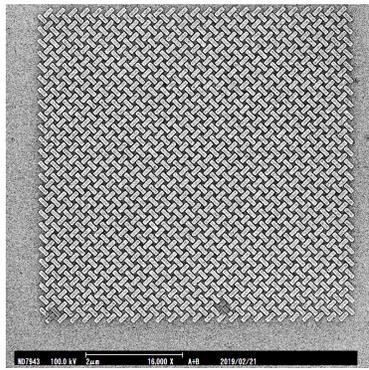
If the shape of the nanomagnet itself is below some critical dimensions (length, width, and thickness) the internal spin moments of the magnet will align in one direction, at a macro level the magnet can then be viewed as a binary element, i.e., the spin points in one of two possible directions along the elongated shape of the magnet. Interactions between a global field or other magnets can cause a magnet to change its state (flip). The arrows in the upper square of Fig. 1a illustrate the alignment of the magnets' spins in one of the two possible directions.

From its origin in experimental physics, there are established lithographical fabrication processes for ASI. Fig. 2a illustrates a produced ASI sample from our ongoing experiments with Pinwheel geometries and Fig. 2b shows an example of a preliminary produced sample of one of our evolved geometries. For experiments targeting behaviour there are imaging techniques that can capture the magnetisation of individual nanomagnets.

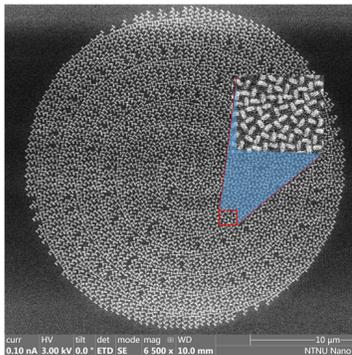
Detailed simulation frameworks like the mumax<sup>3</sup> micro-magnetic simulator (Vansteenkiste et al., 2014) give an accurate and detailed understanding of the internal behaviour of nanomagnetic elements and their interactions (in the range of hundreds of nanomagnets). In other large scale simulators, such as flatspin (Jensen et al., 2020), such fine grained detail is traded off for speed and efficiency. This offers insight into the dynamics of ASI systems at a scale (millions of nanomagnets) where emergence and self-organisational properties are observable in reasonable simulation time. As such, ASI, as a substrate for computation, offers models from interacting atomic spins to large scale meta-material simulations, as well as the opportunity to produce physical samples for experimental verification.

### ASI as computer

Vast parallelism with emergent and self-organisational properties, arising from local communication make ASI a promising substrate for computation. Moreover, ASIs are capable of a huge range of behaviours, just the small adjustment from Square (Fig. 1a) to Pinwheel (Fig. 1b) (elements rotated 45°) gives the system completely different emergent properties. However, to provide useful computation, the system must be tunable to a dynamic regime where computation can be observed and exploited. Recently several promising results demonstrate that ASI is a substrate with such a potential. Within the Reservoir Computing (RC) framework (Jaeger, 2001; Maass et al., 2002) dynamics in Square ASI geometries have been tuned by external magnetic fields to different modes of computation, i.e., memory, classification, or systems including both properties (Jensen et al., 2018). Furthermore, ASI geometries Pinwheel (Jensen and Tufte, 2020) and Kagome (Hon et al., 2021) have been demonstrated to exhibit dynamics that show good scores on standard RC benchmarks of kernel-quality and generalisation-capability (Legenstein and Maass, 2005). The given example of computing in ASI uses external magnetic fields and physical parameters such as distance between nanomagnets as tuning parameter whilst the geometry was fixed.



(a) Manufactured Pinwheel



(b) Manufactured evolved geometry

Figure 2: Scanning electron microscopy (SEM) images of (a) a Pinwheel ASI, and (b) an ASI found through evolution using our ASI representation.

## Representation and evolutionary setup

An EA consists of a population of individuals, a means to discover new individuals (mutation or crossover), and a mechanism to remove less fruitful individuals (selection). Consequently, it also requires some notion of ordering on the individuals, defining which is the most or least fruitful (fitness function). In evolutionary terms, this is the fitness of an individual. Here, we detail our novel representation of individuals for use in an EA, how they are used to generate ASI geometries, how they can be mutated and combined to explore new geometries, and how fitness can be assessed.

## Representation of ASIs

Given that ASIs of interest can be made up of thousands of magnets or more, it would be difficult to use a representation where each magnet is parameterised independently. It follows therefore, that a good representation is one where a few parameters can determine the layout of many magnets, via some genotype to phenotype mapping. A benefit of this approach is that the same representation could be used to generate a given geometry at different sizes.

Common ASI geometries can be scaled to any size

through a simple tiling process using a square tile or unit cell, though such a tiling process would bias the complexity required to create certain geometries of seemingly equal structural complexity, e.g., a Square or Pinwheel geometry can be constructed with a tile of two magnets whereas a Kagome geometry requires three. Ideally a representation for such fundamental geometries as these should, itself, be fundamental. This issue can also be seen in Lindenmayer systems (L-systems) which are commonly employed for generating structures. Where a simple structure does not necessarily have a simple representation. Furthermore, L-systems are very sensitive to small changes in the rule-set (genotype), leading to large changes in the produced structure (phenotype). Whereas, when using an EA, the ability to move smoothly and incrementally between different solutions is highly desirable.

Our representation of an ASI uses a set of tiles, but does not place them in a conventional manner. Each tile consists of exactly two magnets: one ‘origin magnet’ fixed in the centre with  $0^\circ$  rotation, and one ‘free magnet’ that can be positioned anywhere in the tile and have any rotation. A tile can be ‘applied’ to a magnet to generate a new magnet as follows: given a magnet  $M$  and a tile  $T$ , rotate and translate  $T$  such that the origin magnet in  $T$  is completely aligned with  $M$ , then the position and rotation of the free magnet in  $T$  is used to generate a new magnet in the pattern. A full, rigorous description of this is given in algorithm 1, and a pictorial description is given in Fig. 3. Essentially, we use our tiles more as stencils: aligning a tile to a magnet in the system shows where the next magnet can be ‘drawn’. Note that a tile can be applied to the same magnet twice, as the magnets have 2-fold rotational symmetry.

An additional level of complexity can be achieved through attributing symbols to each magnet, similar to that of a rewriting system. When using symbols, in order for a tile  $T$  to be applied to a magnet  $M$ , the symbol of  $M$  must match the symbol of the origin magnet in  $T$ . These symbols can be parameterised, as in parameterised L-systems. In this work, we choose to modify the symbol of a magnet as a function of its current rotation. Specifically, we use a piece-wise function to select the symbol of a magnet depending on whether its orientation is greater or less than  $180^\circ$ . This is by no means the most sophisticated, or likely the best way, to parameterise the symbols. It is merely a simple example we found sufficient to increase the ‘context-sensitivity’ of the generative process.

Our ASIs are built through the iterative application of tiles on magnets, as described above. As such the tiles are analogous to the genes in a genome, a finite number of building blocks which are used to construct an ASI geometry (the phenotype). The process is described in detail in algorithm 2.

As can be seen in algorithm 2, magnets are not placed if they would overlap with an already placed magnet (to en-

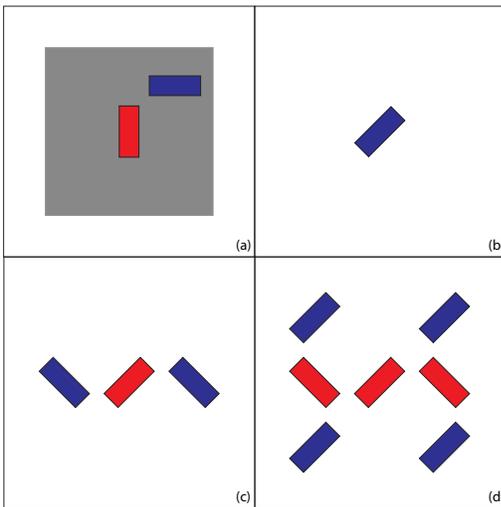


Figure 3: Two applications of a tile, starting from a single magnet. (a) The tile that will be applied, with the origin magnet marked in red. (b) An initial single magnet that the tile shall be applied to. (c) The result after one application of the tile. (d) The result of again applying the tile to the result in (c). At each application, the origin (red) magnet of the tile is overlapped with all magnets which have not yet had a tile applied to them (blue). Continuing the shown iteration produces the Square geometry.

sure physical restrictions of manufacturing are not violated), or if the compared symbols are different. It is also possible to specify the minimum distance permitted between magnets (this is useful for when it comes to physically producing the ASI). Due to these constraints, it is possible that, after a number of steps, no new magnets can be placed. Thus for some tile-sets, the size of phenotype they can produce is bounded. Depending on the use-case of the representation this could be a drawback if geometries of a certain size are required, as some individuals may have to be culled or otherwise if they cannot produce geometries which satisfy a size requirement. Alternatively, this can be viewed as a form of self-regulation, giving the representation, and therefore the EA, control over where the geometry should terminate.

### Mutation and crossover

Given our genotype is a collection of tiles, we have two classes of mutation operators. The first class entails adding or removing tiles to the genotype. In the second class are operations which modify individual tiles within the genotype, through rotating or transposing the second magnet within the tile. Similarly, the two crossover operators produce new individuals by either sampling tiles from two parent individuals, or by combining the angles and positions from the parents' tiles.

---

### Algorithm 1 Apply a Tile to a Magnet

---

```

1: function APPLYTILE(tile, magnet)
2:   newMagnets  $\leftarrow$  empty list
3:   origin  $\leftarrow$  origin magnet of tile
4:   Rotate tile about centre of origin such that ROTATION(origin) = ROTATION(magnet)
5:   for  $\theta \in \{0, 180\}$  do
6:     Rotate tile  $\theta^\circ$  about centre of origin
7:     if SYMBOL(origin) = SYMBOL(magnet) then
8:       add copy of the non-origin magnet in tile to newMagnets
9:     end if
10:  end for
11:  return newMagnets
12: end function

```

---

### Simulating geometries

Using flatspin we can provide input and stimuli to the magnetic system through applying and altering the properties of a global magnetic field. There are limitless options when deciding how an external field can be employed to encode input. Here, for simplicity, we constrain ourselves to applying a global field as a sine wave, and alter the amplitude and the angle at which the field is applied in order to stimulate the system.

flatspin models the time evolution of the system, arising from magnet-to-magnet interactions or influence from a global field, on a flip by flip basis. As such flatspin provides us with a detailed time series of the system's dynamics. At each point in the time series we can view the current spin of each magnet. The ensemble of these binary spin values is what we define as the state of the system.

### Evolving geometries

In this section we employ two fitness functions to assess the representation's ability to facilitate the evolutionary search for a material property: the resilience to an external field, and a computation property: the number of unique states in the system's trajectory.

#### Resilience: Minimising flipping and examining scalability

Here we search for a simple, yet non-trivial property of an ASI; the resilience of an ASI to an external field. That is, under a series of global field applications, find an ASI which minimises the number of magnet flips that occur. To achieve our goal of resilience, we use a minimum flips fitness function. Secondly, to demonstrate scalability in our representation, we show that we can search for a property in a ASI of one size, and then scale up the chosen ASI and see it retains the desired property.

If an oscillating field was applied at only one angle, then given the elongated, rectangular shape of our magnets the

---

**Algorithm 2** Produce ASI from a tile set

---

```
1: function GENERATEASI(tileSet, maxSize)
2:   frontier  $\leftarrow$  empty list
3:   frozen  $\leftarrow$  empty list
4:   insert into frontier a magnet with position=(0, 0), angle=0 and symbol = 0
5:   while SIZE(frontier) + SIZE(frozen) < maxSize and SIZE(frontier) > 0 do
6:     newMagnets  $\leftarrow$  {APPLYTILE(tile, magnet) |  $\forall$  tile  $\in$  tileSet,  $\forall$  magnet  $\in$  frontier}
7:     remove all magnets in frontier and add them to frozen
8:     for magnet  $\in$  newMagnets do
9:       if magnet does not intersect any magnet in frozen then
10:        add magnet to frontier
11:       end if
12:     end for
13:   end while
14:   append magnets in frontier to frozen
15:   return TRUNCATE(frozen, maxSize)
16: end function
```

---

solution would trivially be to orient all magnets orthogonal to the field, aligning it to their hard axis. We observed such behaviour in preliminary evolutionary runs of this fitness function. To increase the difficulty of the task, and thus give a more interesting solution, we oscillate a field at  $0^\circ$  and then at  $90^\circ$ . Now the EA must take account of the local interactions between the magnets rather than only the global property of a magnet's angle compared to the angle of the global field.

As we are evolving the geometry of the ASI and not the initial state, we do not want the fitness to heavily depend upon the initial state, that is, the starting orientation (spin) of magnets in the geometry. If the magnets happen to be initialised in a state of high energy they are more likely to flip. To remedy this, we cycle the fields multiple times and do not record flipping that happens in the first field cycles to allow the system to settle into a lower energy state, as to not penalise unfavourable initial states.

A benefit of the scalable nature of our representation is that it allows us to find candidate ASIs that would otherwise be of too greater size to search for in reasonable time. For some given property, we can search more quickly for an ASI at a smaller size. When sufficiently fit individual is found, we can use the representation to grow it to a larger size and evaluate if it still retains the desired property. Of course, the fitness of this individual may strongly depend on the size to which it is grown. To encourage the EA to produce individuals with fitness less sensitive to their size, we evaluate their fitness at multiple sizes.

Geometries of 100 magnets were evolved using the following evolutionary parameters: populations size = 200, maximum number of generations = 500, mutation rate = 0.3 and crossover rate = 0.2. Here we allow individuals to consist of up to two tiles, and a symbol alphabet of size 3. The magnets have dimension  $220\text{ nm} \times 80\text{ nm}$  and

cannot be placed closer than 20 nm to another. Under this fitness function, it would be advantageous for an individual to self-regulate itself into producing less than 100 magnets (fewer magnets, fewer flips). Thus individuals that cannot produce the required number of magnets are not evaluated, and assigned the worst possible fitness (though they are still able to undergo mutation and crossover). Roulette selection with elitism is used to remove low fitness individuals from the population.

Geometries are mapped into flatspin using the following flatspin parameters:  $hc = 0.2$ ,  $\alpha = 30272$ ,  $sw_b = 0.4$ ,  $sw_c = 1$ ,  $sw_\beta = 3$ ,  $sw_\gamma = 3$ . The deceptively large magnitude of  $\alpha$ , the flatspin parameter specifying the strength of interaction between individual magnets, is due to the difference of units used in the tiles and flatspin (nanometers instead of meters). The fitness is evaluated by applying one period of a sinusoidal field, with amplitude  $H$ , at  $0^\circ$  then again at  $90^\circ$ . This is repeated four more times and the total number of times any magnet flips is recorded, excluding those that flipped in the first two periods. As aforementioned, we also apply this procedure to smaller fractions of the geometry. We evaluate the number of flips recorded when growing the individual to 20, 40 and 80 magnets, as well as full size (100 magnets). Thus our minimum flips fitness function is given by the sum of these values, and it is this that the EA attempts to minimise.

Fig. 4 shows separate runs of the EA with different values of field strength  $H$ . Clearly the difficulty of the task to find a geometry that minimises the number of flips is heavily dependent on  $H$ ; for a weak field it is easy to choose a geometry with few or no flips. To illustrate the inherent difficulty at each field strength a random sample of 100 individuals was evaluated with the minimum flips fitness function, and show the spread of these results. If a randomly generated individual cannot meet the geometry size requirement due to

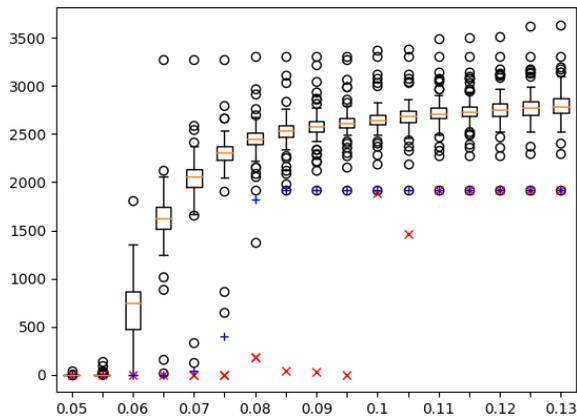


Figure 4: Box plot showing the spread of the minimum flips fitness when applied to 100 randomly initialised geometries, at each field strength  $H$ . Red crosses indicate the fitness of best geometry discovered through evolution. For reference we also include a perfect Square geometry (blue pluses) with the smallest magnet-to-magnet distance that was available to the EA. This was also done for Pinwheel and Kagome, but they are excluded from the graph as they were strictly outperformed by Square.

self-regulation, it is discarded and re-sampled.

From Fig. 4 we see that the EA performs as well as or better than the best individual in the random sample at all values of  $H$ , and significantly better for some values of  $H$ . At low values of  $H$  it is unlikely that flips occur regardless of the geometry, we see the random sample producing many low fitness individuals. Towards the higher field values, we see saturation in the minimum fitness that can be produced in both the results of the EA and the random sample. At these high field values, it appears that the solution of least flips resembles the trivial, single field direction case, in that the magnets are orientated close to perpendicular against one of the fields thus only flipping every other field cycle. This is made clear by the minimum fitness saturating at 1920 exactly half of the maximum possible fitness.

The area possibly of most interest in Fig. 4 is the region  $H \in [0.08, 0.095]$ , where the field is strong enough that all geometries in the random sample have high fitness, but the EA is still able to achieve low, and in some cases, perfect (0) fitness. The fitness in this region being well below 1920, while the random sample has saturated. This indicates that the EA is finding geometries which exploit the interaction between magnets to prevent the flipping, rather than considering only each individual magnets orientation with respect to the global field angle. Surprisingly, we see the best fitness found at  $H = 0.08$  (117) is noticeably worse than the best found at  $H = 0.095$  (0). In fact, the individual found at  $H = 0.095$  has perfect fitness when evaluated at  $H = 0.08$ . This discrepancy is likely due to the inherent randomness of

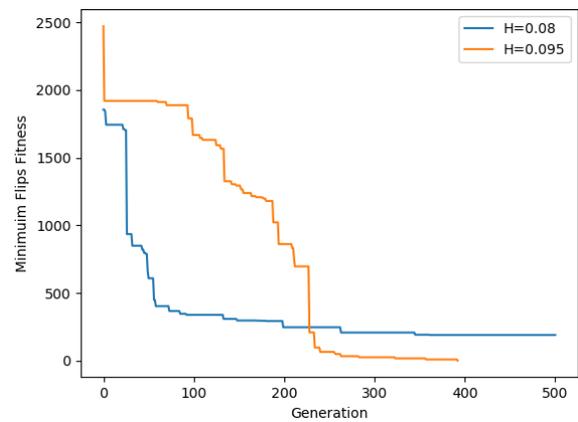


Figure 5: Graph showing the minimum flips fitness of the best individual in the population at each generation, for  $H = 0.08$  and  $H = 0.095$ .  $H = 0.095$  terminates early as it achieves perfect fitness (0)

the evolutionary search; looking at Fig. 5 we can see that a much steeper descent in fitness in the  $H = 0.08$  case. This seems to indicate the poorer results are the product of premature convergence and could be remedied by diversity measures or otherwise, though such things are outside of the scope of this work.

In Fig. 6 we have taken the geometry found by the EA at  $H = 0.07$  and evaluated its fitness after scaling it up to different sizes. We see at a 1.5 scaling factor the perfect fitness of the individual is retained; at 2 and 4 times the fitness is still relatively low. We see a degradation of the scaling above this point with the dramatic increase in fitness at 8 times scaling.

### Diversity: State count novelty search

Now we examine the EA's performance when using a computational property as our fitness function. Commonly, in the pursuit of computation with a dynamical system, the trajectory of the system is leveraged for computation e.g., ballistic computing (Stepney, 2008) or RC. Such approaches require the ability to perturb the trajectory via some input to the system.

Here we explore the ability of our representation to produce geometries with variety of responses to an input, and the ability of the EA to locate them. Specifically, we define the property 'state count' as the number of unique states an ASI passes through as it is perturbed by a series of inputs. We fix the input series and use a naïve novelty search fitness, attempting to find geometries with state counts not previously seen by the EA. Essentially, we are aiming to maximise the number of different state counts present in the population referred to as the state count diversity. The state count diversity is therefore maximised when the population contains an individual for every possible value of state count.

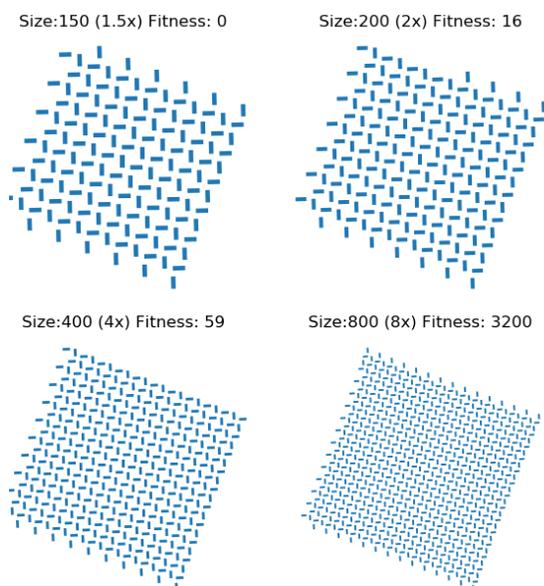


Figure 6: The best geometry found for the minimum flipping fitness function at  $H = 0.07$  scaled up to 1.5, 2, 4 and 8 times its original size.

We encode an input bit string as a variation on the angle of a global sinusoidal field acting upon the system. A zero or one in the input bit string represents applying a field cycle at  $0^\circ$  or  $90^\circ$  respectively. After each field application the state of the system is recorded. As such, the maximum number of unique states observed under an input string of length  $n$  is

The EA and flatspin parameters remain almost unchanged from previous experiment, the only difference is that we increase both the number of tiles and symbol alphabet size to 6. This allows more complex geometries to be found, and vastly increases the search space, thus allowing for greater diversity in the population. We fix the input to the arbitrarily chosen bit string “0101 0011 0011 0000 1111”. Geometries are simulated in flatspin and their state count is computed. The fitness of an individual, which the EA is minimising, is the difference between the state count of the individual and the closest state count value not yet found by the EA. Though this is a rather crude implementation of novelty search, given the very small and discrete behavioural space in this case, we believe it is sufficient. As before, individuals which cannot meet the geometry size requirement receive the worst possible fitness score.

Again, as our problem of interest is likely dependent on the field strength  $H$  of our input to the system, we test the EA at multiple values of  $H$  and compare it to random samples to gain some intuition of the difficulty of the task. As state count diversity is a property of the population not an individual, we must sample random populations. 100 independent populations containing 100 randomly initialised

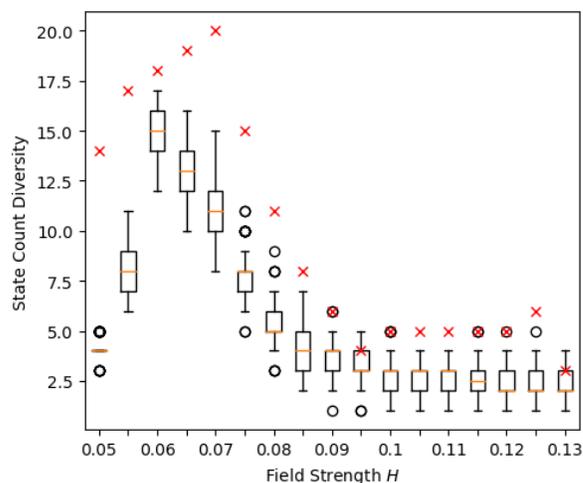


Figure 7: Box plot showing the spread of the unique states fitness when applied to 100 randomly initialised geometries. The red crosses indicate the number of different state counts the EA was able to discover, 20 being the maximum possible.

geometries were generated for each value of  $H$ . Then, for each population, the state count diversity was calculated. In Fig. 7 we see that, for all but two  $H$  values, the EA finds an as good or better variety of geometries *w.r.t* state count. At  $H = 0.07$  the EA achieves perfect state count diversity, meaning for any desired state count we can supply a geometry which provides it. We see two instances towards the end of the graph where the EA fails to find as many different state counts as seen in the best populations of the random sample. These instances both occur after the variety of different state counts has begun to saturate, and in both cases the EA achieves a state count diversity one fewer than the best population in the sample.

## Discussion

Through the use of an EA, facilitated by our novel ASI representation, we were able to discover geometries resilient to external influence. More generally, this shows this representation can be used as a useful tool when searching for a structure with certain, emergent, physical properties.

In our experiment on scalability we observed perfect scaling on the examined geometry at 1.5 scaling and a gentle degradation fitness as the scaling increased up to 4. Clearly there is some threshold between 4 and 8 at which the fitness suffers greatly from increases in size, and the exact location of this threshold is likely strongly tied to the properties of the evolved individual. The apparently poor scalability at the highest scaling factor is somewhat to be expected, after all it is by no means uncommon for extrapolations to fail when too far from the original measurements. Such scaling problems could arise from small numerical noise in the

position and angle of magnets, which is amplified when the geometry is scaled up. Despite this, the results show our representation has potential for scalability, and we expect the scalability observed here could be improved upon with more a sophisticated approach to encouraging scalability in the geometries. As an example, rewarding the geometries which exhibit higher levels of self-similarity could improve scalability. Alternatively, Harding and Miller (2007) demonstrate the merit in re-evolving individuals when necessary; one could envision a process in which an EA is run quickly for many generations with geometries of a smaller sizes, then, the best individuals could be scaled up and undergo evolution for a small number of generations in order to fine-tune them.

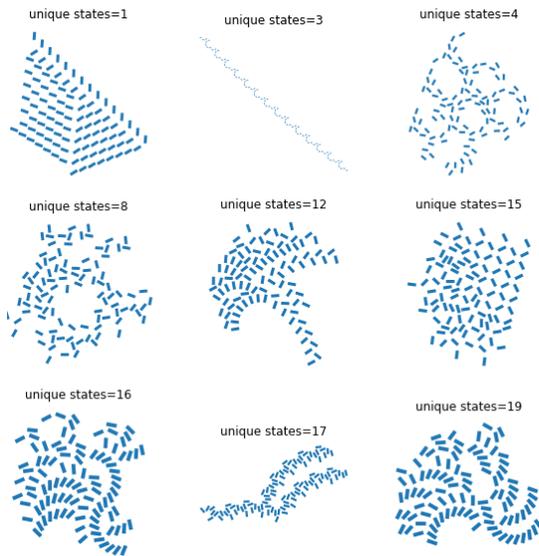


Figure 8: A selection of the geometries found in the search for different unique state counts at  $H = 0.07$

The novelty search targeting the state count property demonstrated the representation’s capability in producing diverse geometries with a variety of responses to a given input. Furthermore, through purely visual examination of the structures shown in Fig. 8, we can see these different individuals vary greatly in their structure also. Such a range is crucial to a representation if it is to be exploited for evolutionary search, particularly when we have little knowledge a priori on the kind of geometry that would perform well on a given task.

In both evolutionary runs, we see good variety of different structure being explored, indicating the range representation covers sufficiently large region of the search space. Furthermore, from the  $H = 0.095$  fitness evolution shown in Fig. 5, we see the EA is capable of moving smoothly through the behavioural landscape using our representation. Of course, premature convergence will always be an issue when there are no measures in place to combat it.

With this representation, many of the most well-studied and simple geometries, e.g., Square, Pinwheel and Kagome, are represented by the simplest form of individual - an individual consisting of only one tile. Consequently, in this representation, the transition between these geometries can be completely continuous. More complex geometries arise from individuals made up of more tiles, which gives a mechanism to bias the complexity of individuals produced in an evolutionary search.

Engel et al. (2018) use machine learning to generate a large zoo of possible ice structures. Their search is bounded by thermodynamic constraints. In ASI we have no such constraints, which may be why no similar attempt has been made for ASI. With our representation such an endeavour may now be feasible, as it give a way to generate many geometries and allows restrictions to be placed on the geometries size or complexity. In a similar vein, one could classify geometries based on their computation properties. Dale et al. (2019) give a framework which can be used to measure the potential of a substrate for use in RC, and to indicate which computational tasks the substrate would excel at. Using such a framework in conjunction with our representation, would allow the creation of a catalogue of geometries sorted by the class of computational task they perform well in.

## Conclusion

In this paper we demonstrate what is, to the best of our knowledge, the first example of the computational properties of ASI being tuned through the changing of geometry. This is made possible by our novel representation of ASI geometry, which provides an EA with a mechanism to generate many varied geometries. Such a variety is key when the relation between system configuration and the emergent behaviour is mostly unknown. The representation’s ability to move smoothly and incrementally through the search space allows the EA to follow gradients in the fitness landscape.

Scalability arises naturally from the generative process at the heart of our representation. Our results showed geometries can be tuned at one size and then scaled up, while retaining their observed behaviour. Though we note the failure of this scaling above a certain threshold, we provide some viable solutions to further extend this threshold.

The domain of ASI geometries is severely underexplored, with research mostly split between basic tilings and, at the other end of the spectrum, heavily engineered nanomagnetic logic systems. Our hope is that this representation opens the door to a middle ground of complex, interesting and unforeseen geometries with computationally useful properties.

**Acknowledgements:** This work was funded in part by the Norwegian Research Council IKTPLUSS project SOCRATES (Grant no. 270961), and in part by the EU FET-Open RIA project SpinENGINE (Grant no. 861618). Simulations were executed on the NTNU EPIC compute cluster (Själänder et al., 2019).

## References

- Bose, P. (2011). *Power Wall*, pages 1593–1608. Springer US, Boston, MA.
- Dale, M., Miller, J. F., Stepney, S., and Trefzer, M. A. (2019). A substrate-independent framework to characterize reservoir computers. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 475(2226):20180723.
- Dennard, R. H., Gaensslen, F. H., Rideout, V. L., Bassous, E., and LeBlanc, A. R. (1974). Design of ion-implanted mosfet's with very small physical dimensions. *IEEE Journal of Solid-State Circuits*, 9(5):256–268.
- Downing, K. (2005). Tantrix: A minute to learn, 100 (genetic algorithm) generations to master. *Genetic Programming and Evolvable Machines*, 6:381–406.
- Engel, E., Anelli, A., Ceriotti, M., Pickard, C., and Needs, R. (2018). Mapping uncharted territory in ice from zeolite networks to ice structures. *Nature Communications*, 9.
- Harding, S. L. and Miller, J. F. (2007). Evolution in materio: Computing with liquid crystal. *Journal of Unconventional Computing*, 3(4):243–257.
- Hon, K., Kuwabiraki, Y., Goto, M., Nakatani, R., Suzuki, Y., and Nomura, H. (2021). Numerical simulation of artificial spin ice for reservoir computing. *Applied Physics Express*, 14(3):033001.
- Jaeger, H. (2001). The “echo state” approach to analysing and training recurrent neural networks - with an Erratum note. Technical Report 148, German National Research Center for Information Technology.
- Jensen, J. H., Folven, E., and Tufte, G. (2018). Computation in artificial spin ice. *The 2018 Conference on Artificial Life: A Hybrid of the European Conference on Artificial Life (ECAL) and the International Conference on the Synthesis and Simulation of Living Systems (ALIFE)*, (30):15–22.
- Jensen, J. H., Strømberg, A., Lykkebø, O. R., Penty, A., Sjalander, M., Folven, E., and Tufte, G. (2020). Flatspin: A Large-Scale Artificial Spin Ice Simulator. *arXiv:2002.11401 [cond-mat, physics:physics]*.
- Jensen, J. H. and Tufte, G. (2020). Reservoir computing in artificial spin ice. *Artificial Life Conference Proceedings*, (32):376–383.
- Legenstein, R. and Maass, W. (2005). What makes a dynamical system computationally powerful? In *New Directions in Statistical Signal Processing From Systems to Brain*, pages 127–154.
- Maass, W., Natschläger, T., and Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531–2560.
- Macêdo, R., Macauley, G. M., Nascimento, F. S., and Stamps, R. L. (2018). Apparent ferromagnetism in the pinwheel artificial spin ice. *Physical Review B*, 98(1):014437.
- Miller, J. F., Harding, S., and Tufte, G. (2014). Evolution-in-materio: evolving computation in materials. *Evolutionary Intelligence*, 7(1):49–67.
- Monroe, D. (2014). Neuromorphic computing gets ready for the (really) big time. *Commun. ACM*, 57(6):13–15.
- Moore, G. E. (1965). Cramping more components onto integrated circuits. *Electronics*, 38(8).
- Sjalander, M., Jahre, M., Tufte, G., and Reissmann, N. (2019). EPIC: An Energy-Efficient, High-Performance GPGPU Computing Research Infrastructure. *arXiv:1912.05848 [cs]*.
- Sklenar, J., Lao, Y., Albrecht, A., Watts, J. D., Nisoli, C., Chern, G.-W., and Schiffer, P. (2019). Field-induced phase coexistence in an artificial spin ice. *Nature Physics*, 15(2):191–195.
- Stepney, S. (2008). The neglected pillar of material computation. *Physica D: Nonlinear Phenomena*, 237(9):1157–1164. Novel Computing Paradigms: Quo Vadis?
- Teuscher, C. and Adamatzky, A. (2005). *Unconventional Computing 2005: From Cellular Automata to Wetware*. Luniver Press.
- Turing, A. M. (1937). On Computable Numbers, with an Application to the Entscheidungsproblem. In *Proceedings of the London Mathematical Society 1936-37*, volume 42 of 2, pages 230–265. London Mathematical Society.
- Vansteenkiste, A., Leliaert, J., Dvornik, M., Helsen, M., Garcia-Sanchez, F., and Van Waeyenberge, B. (2014). The design and verification of MuMax3. *AIP Advances*, 4(10):107133.
- von Neumann, J. (1945). First draft of a report on the edvac. edited by m. d. godfrey 1992. Technical report, Moore School of Electrical Engineering University of Pennsylvania.
- Wang, R. F., Nisoli, C., Freitas, R. S., Li, J., McConville, W., Cooley, B. J., Lund, M. S., Samarth, N., Leighton, C., Crespi, V. H., and Schiffer, P. (2006). Artificial ‘spin ice’ in a geometrically frustrated lattice of nanoscale ferromagnetic islands. *Nature*, 439(7074):303–306.