Scalable Evolution of Logically Independent Polycomputational Materials

Piper Welch[1(✉)], Atoosa Parsa[2], Shawn Beaulieu[1], Corey S. O'Hern[3], Rebecca Kramer-Bottiglio[3], and Josh Bongard[1]

[1] Department of Computer Science, University of Vermont, Burlington, VT 05405, USA
{piper.welch,shawn.beaulieu,josh.bongard}@uvm.edu
[2] Department of Biology, Tufts University, Medford, MA 02155, USA
atoosa.parsa@tufts.edu
[3] Department of Mechanical Engineering and Materials Science, Yale University, New Haven, CT 06520, USA
{corey.ohern,rebecca.kramer}@yale.edu

Abstract. Substrates that compute using vibration rather than electricity offer the potential of creating and deploying computers into electronics-denying environments. Another advantage of these materials is that some of them can compute multiple functions in the same place and at the same time, providing computational results at different frequencies. These so-called polycomputational materials may eventually compete with more traditional computers in terms of computational density because there is no currently known upper bound on how many functions can be simultaneously computed by a vibrational substrate. However, three challenges remain for polycomputational materials: how to ensure that the different functions are computed independently; developing evolutionary algorithms that allow for embedding increasingly more functions into these materials *in silico*; and validating the evolved *in silico* materials as physical materials. Here we report progress on all three of these issues.

Keywords: Granular Metamaterials · Mechanical Computing · Polycomputing

1 Introduction

The demand for faster and more efficient computation continues. Since the 1950 s, silicon-based computers have dominated. However, these conventional systems fail in harsh environments, rely on electricity, and degrade into electronic waste [8,19]. Unconventional computers are those which explore novel substrates and bit abstractions [5,7,9,15,16,20]. They offer alternatives that are more resilient in extreme conditions, operate without traditional electricity, or produce less electronic waste [4,7,11,21]. Despite these potential advantages, unconventional computers currently fall far short of the computational ability and density of digital computers. However exponential growth patterns such as

P. García-Sánchez et al. (Eds.): EvoApplications 2025, LNCS 15612, pp. 558–572, 2025.
https://doi.org/10.1007/978-3-031-90062-4_35

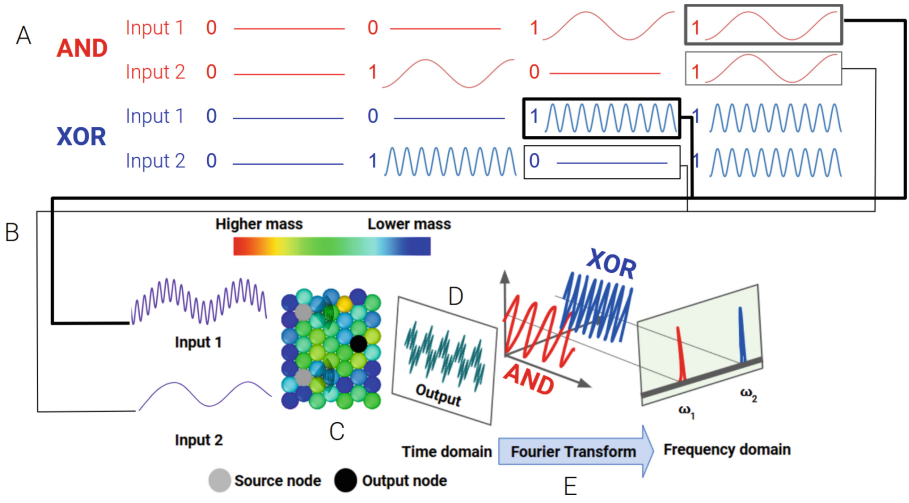

Fig. 1. CGMM inputs and outputs. This figure shows an overview of the system inputs and outputs for a CGMM where gate g_1 = AND and g_2 = XOR. **A)** This panel shows the possible system inputs for the AND gate operating at ω_1Hz and the XOR gate operating at ω_2Hz. In this example, the '11' input is passed into the AND gate, and the '10' input is passed to the XOR gate. **B)** The input signals are a combined superposition of the individual input cases. **C)** The CGMM is supplied with the selected input cases. In this cartoon, the input grains are marked in gray, while the output grain is marked in black. **D)** The output signal in the time domain. **E)** We perform a fast Fourier transformation on the time-domain oscillation of the output grain. We then analyze the power at the driving frequencies where power above a given threshold corresponds to an output value of '1' and power below a given threshold corresponds to an output value of '0'.

Moore's Law are not sustainable indefinitely. Transistors are nearing their physical limits, which impedes further miniaturization [17]. The particular class of unconventional computing material investigated herein, so-called polycomputational materials, can compute different logic gates at different frequencies [2,3]. Since there are infinite frequencies, in theory, such materials could have infinite computational density. In practice, no upper bound on computational density has been determined for these materials.

Polycomputational materials are designed using machine learning to engineer the properties, such as stiffness or mass, of grains in granular metamaterials [10]. Such materials are referred to as *computational granular metamaterials* (CGMMs) [1,12–14]. An overview of the inputs and outputs of a CGMM is displayed in Fig. 1. In addition to offering potentially unbounded computational density, CGMMs offer several advantages over traditional computational architectures. Notably, they could provide more robust and energy-efficient computation, as they would not rely on electronic components or specific substrates. These systems also have the potential to revolutionize future robotic systems, enabling sensing, control, and actuation to be driven by vibrational computation.

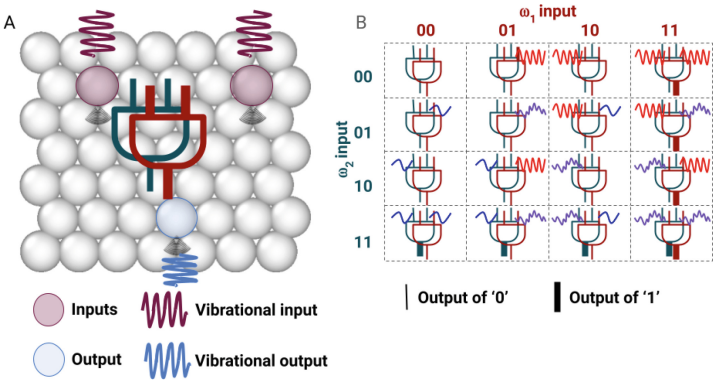

Fig. 2. Logical independence of a CGMM. A) Overview of a granular material with superimposed AND gates g_1 and g_2 operating at ω_1 and ω_2 hertz respectively. Bits are treated as the presence (1) or absence (0) of vibration at a given frequency. In panel a), both input grains are vibrated at ω_1Hz, representing 1,1 supplied to gate g_1. Neither is vibrated at ω_2Hz, representing the simultaneous application of 0,0 to gate g_2. The output grain is observed to vibrate in response, correctly, at ω_1Hz (1,1→1), but not vibrate, correctly, at ω_2Hz (0,0→0). **B)** For a given material, one can ask how many of the 16 possible cases (each of the four input cases, at both frequencies) results in the correct behavior of the output grain. Here, both the logic gates within the material are AND. An ideal material is logically independent: the gate at one frequency acts correctly regardless of which of the four input cases is being supplied to the other gate at the other frequency.

As polycomputational granular materials are an emerging technology, we have yet to explore many of their properties. One such unexplored yet important behavior is the relative *independence* of gates embedded within a given substrate. Here, independence refers to the ability of a logic gate to function autonomously, regardless of the input received by other gates within the same substrate. Prior studies have concentrated on the behavior of these materials across the input scenarios where source grains receive identical logical inputs *simultaneously* (e.g., the gate operating at frequency ω_1 receives the input case '10', while the gate operating at frequency ω_2 also receives the input case '10') [12–14]. Therefore, it remains unknown if gates g_1 and g_2 in a given material exhibit any degree of independence. That is to say, we do not have insight into how much the logical input into g_1 impacts the behavior of g_2 and vice versa.

One might think a simple solution to achieving logically independent CGMMs is to evolve for proper behavior at each of the 16 input case pairs (shown in Fig. 2B). However, explicitly evolving a granular material for independence has poor scalability: as logic gates n embedded into a material increases, the potential superpositions of frequencies expand exponentially with $O(2^{2n})$. Therefore, in this work, we focus on the following questions: 1) Can we evolve logically independent CGMMS, and 2) how do we do so in a scalable manner? To answer these questions, we evolve CGMMs using a variety of methods. We find that

it is possible to evolve independent CGMMs and we can do so using scalable methods. This work presents the first CGMMs known to have this property and therefore highlights the potential of CGMMs to serve as dependable and robust computational systems.

In the following section, we describe our *in silico* model of granular materials, and their physical environments, followed by an outline of our evolutionary methods and conducted experiments. The Results section states our findings. The Discussion Section enumerates the behaviors of the best-evolved materials and their relative independence. The Hardware Implementation section presents a minimal physical implementation of a granular logic gate. We conclude with future directions for this work, including the transfer of our evolved designs *in situ*.

2 Methods

We use multi-objective optimization to evolve the masses of grains in a granular material. Each material has two logic gates, gates g_1 and g_2 operating at frequencies ω_1 and ω_2. For each material, either g_1 = AND and g_2 = AND, or g_1 = XOR and g_2 = AND. To explore scalable methods of evolving independent CGMMs, we test four methods of evaluating material fitness during evolution. Specifically, in place of evaluating materials for correct behavior in each of the 16 combinations of input case pairs, we evaluate gate behavior on various subsets of input case pairs for input$_1$ at $\omega_1 \in$ ['00', '01', '10', '11'] and input$_2$ at $\omega_2 \in$ ['00', '01', '10', '11']. These differing input case pair subsets used during evolution are referred to as *evaluation methods*. 30 independent evolutionary runs were collected for each experiment. We use the Wilcoxon rank-sum test and Bonferroni corrections for all statistical analyses with multiple pairwise comparisons.[1]

2.1 CGMM Definitions

In this work, a material consists of 49 3D grains arranged in a seven-by-seven triangular lattice. They are placed within a simulation box featuring fixed boundaries along the x and y directions. Each grain shares the same diameter D = 10cm. Logical inputs are sinusoidal vibrations supplied to two fixed input grains, and outputs are vibrations measured at a fixed output grain. The locations of output and input grains are arbitrarily fixed as the grains marked in Fig. 2. The logical input of True, or '1', is encoded as a sinusoidal input of amplitude $A = 1$ at driving frequency ω_n. Conversely, the logical input of False, or '0', is encoded as a sinusoidal input of amplitude $A = 0$ at driving frequency ω_n. To quantify output signals, we perform the fast Fourier transformation to convert output signals from the amplitude-time domain to the power-frequency domain. For a granular material to function as a logic gate, the relative magnitude of the

[1] The source code for the experiments in this paper is available here: https://github.com/piperwelch/logical_independence/.

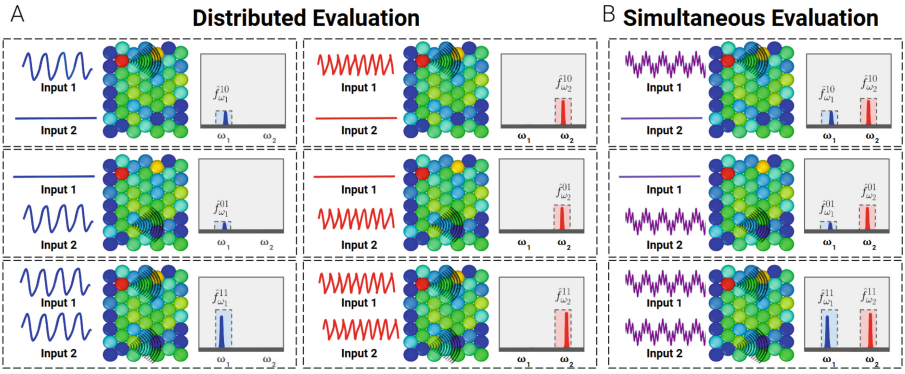

Fig. 3. Comparison of the distributed and simultaneous evaluation methods. A) This panel shows the inputs and outputs for the distributed evaluation method, in which the material is first supplied with the blue frequency, then the red frequency. **B)** This panel shows the inputs and outputs for the simultaneous evaluation method, in which the material is supplied with the red and blue frequencies simultaneously.

power at the driving frequency in each case must be consistent with the desired behavior of the gate. For example, when computing AND, we expect a relatively high power at the driving frequency when both input grains are supplied with vibration. In this work, gate g_1 is optimized to function at 15 Hz, while gate g_2 is optimized to function at 20 Hz.

2.2 Definitions

Notation. Herein, we will refer to a given input case pair supplied to a material as such: $\mathrm{F}_x^{\omega_n}(ij_n)\mathrm{F}_x^{\omega_m}(ij_m)$, where ω_n and ω_m are the driving frequencies, ij_n and ij_m are the logical inputs $\in$ ['11', '01', '10', '00'] supplied in the x-direction at ω_n and ω_m, respectively. For example, we represent the case when '11' is supplied at ω_1 and '00' is supplied at ω_2 as $\mathrm{F}_x^{\omega_1}(\text{'11'})\mathrm{F}_x^{\omega_2}(\text{'00'})$.

Independence. We define the independence of a material as the ability of its embedded gates to function correctly regardless of the input cases applied to other gates within the same material. The metric for quantifying independence is defined as follows:

$$I = \sum_{i=1}^{15} B_i \tag{1}$$

Here, B_i is 1 if behavior i is what we would expect in an ideal logical gate (e.g. if we observe a '1' where we expect it), and 0 otherwise. To determine whether an observed output signal aligns with the expected output, we binarize the material's output signals using a power threshold. This threshold is calculated as the midpoint between the lowest power magnitude corresponding to

an output of 1 and the highest power magnitude corresponding to an output of 0. The independence I is a value $\in$ [0, 15], where 15 represents a material that behaves appropriately in each of the 15 input case pairs. The input case $\mathrm{F}_x^{\omega_1}(\text{`00'})\mathrm{F}_x^{\omega_2}(\text{`00'})$ is not considered as it is trivial.

2.3 Simulator

In this work, we use the open-source software LAMMPS (Large-scale Atomic/ Molecular Massively Parallel Simulator) [18] to model our granular materials. Both normal and tangential grain interactions adhere to a Hookean contact model. A Hookean model of contact also governs grain interactions with the system boundaries. We use a time step of 0.001 and run our simulations for 2000 steps. All reported units are expressed in SI units.

2.4 Optimization

We use a $(\lambda+\mu)$ multi-objective evolutionary algorithm, where $\lambda = 100$ and $\mu = 200$. The genome of each material is a 49-length array of floats, where each float represents the mass of one grain within the material. Each value in a material's genome is randomly initialized on a uniform distribution $\in$ [0.5 kg, 1.3 kg].

2.5 Fitness

Each material has a fitness associated with gate g_1, and a fitness associated with gate g_2. In practical applications, logic gates are either functional or non-functional; however, during evolution, we use a continuous fitness function to enable evolution to follow a fitness gradient. The metric we use to quantify logical AND behavior, or "ANDness" is:

$$\mathrm{ANDness}_{g_n} = \frac{\hat{f}_{\omega_n}^{\text{`11'}}}{(\hat{f}_{\omega_n}^{\text{`10'}} + \hat{f}_{\omega_n}^{\text{`01'}})/2} \tag{2}$$

while the metric we use to quantify logical XOR behavior, or "XORness" is:

$$\mathrm{XORness}_{g_n} = \frac{(\hat{f}_{\omega_n}^{\text{`01'}} + \hat{f}_{\omega_n}^{\text{`10'}})/2}{\hat{f}_{\omega_n}^{\text{`11'}}} \tag{3}$$

Here, $\hat{f}_{fn}^{ij}$ is the power at the driving frequency of the Fourier transform when logical input ij is supplied to the material. These metrics are maximized when the power for outputs that should be '1' is high relative to outputs that should be '0'. The '00' input case is not included in these equations since it is trivial. These metrics are subsequently referred to as GATEness.

Because power exists on a spectrum, while traditional logic gates operate in a binary manner, we establish a threshold of power post-evolution to binarize the behavior of our materials. Signals above this threshold are interpreted as

'1', while those below are interpreted as '0'. To encourage comparable power magnitudes for output signals from gates g_1 and g_2, we introduce a fitness term p. Here, p is designed to introduce pressure for similar output power when materials produce a '1' output. It is defined as:

$$p = \frac{1}{(1 + |\hat{f}_{f1} - \hat{f}_{f2}|)} \tag{4}$$

The fitness of a material for ω_n is then calculated as:

$$\text{fitness}_{g_n} = \text{GATEness}_{g_n} * \ p \tag{5}$$

2.6 Evaluation Methods

The varying subsets of input case pairs used during evolution to assess the fitness of g_1 and g_2 are subsequently enumerated.

Distributed Evaluation. The first evaluation method we test evaluates gate fitness when input cases are supplied to a material in a distributed manner. In this evaluation method, materials are evolved for correct logical behavior for gates g_1 and g_2 when the inputs for g_1 and g_2 are supplied during different simulations. This means a material is simulated six times to construct fitness_{g_1} and fitness_{g_2}. The input case pairs for distributed evaluation are represented visually in Fig. 3A. In closed form, fitness_{g_1} is constructed by examining the material's behavior when supplied with the $\text{F}_x^{\omega_1}(\text{'11'})\text{F}_x^{\omega_2}(\text{'00'})$, $\text{F}_x^{\omega_1}(\text{'10'})\text{F}_x^{\omega_2}(\text{'00'})$ then $\text{F}_x^{\omega_1}(\text{'10'})\text{F}_x^{\omega_2}(\text{'00'})$ input case pairs. Subsequently, fitness_{g_2} is constructed by examining the material's behavior when supplied with the $\text{F}_x^{\omega_1}(\text{'00'})\text{F}_x^{\omega_2}(\text{'11'})$, $\text{F}_x^{\omega_1}(\text{'00'})\text{F}_x^{\omega_2}(\text{'10'})$ then $\text{F}_x^{\omega_1}(\text{'00'})\text{F}_x^{\omega_2}(\text{'01'})$ input case pairs. The number of simulations this method requires scales as $O(n)$ with the n the number of gates embedded within a material.

Simultaneous Evaluation. The second evaluation method we test evaluates gate fitness *simultaneously*. That is, under this evaluation method, materials are evolved for correct logical behavior for gates g_1 and g_2 when the inputs for g_1 and g_2 are supplied during the same simulation. This means a material is simulated three times to construct fitness_{g_1} and fitness_{g_2}. The input case pairs for simultaneous evaluation are represented visually in Fig. 3B. In closed form, fitness_{g_1} and fitness_{g_2} are constructed by examining the material's behavior when it is supplied with the $\text{F}_x^{\omega_1}(\text{'11'})\text{F}_x^{\omega_2}(\text{'11'})$, $\text{F}_x^{\omega_1}(\text{'10'})\text{F}_x^{\omega_2}(\text{'10'})$ and $\text{F}_x^{\omega_1}(\text{'01'})\text{F}_x^{\omega_2}(\text{'01'})$ input case pairs. The number of simulations this method requires scales as $O(1)$ with the n the number of gates embedded within a material.

Input Case Varying Evaluation. The third evaluation method we test evolves for correct behavior as evaluated using three randomly selected input cases. That is, in this method of evaluation, granular materials are evolved for

behavior at g_1 and g_2 when input cases for g_1 and g_2 are randomly selected $\in$ [‘11’, ‘01’, ‘10’, ‘00’]. The input case pair $F_x^{\omega_1}$(‘00’)$F_x^{\omega_2}$(‘00’) and repeat selection of the same input case pair are excluded from consideration due to their redundancy. Figure 2B visually displays the 15 possible input case pair choices. A new set of three input case pairs is chosen for each generation.

Due to the shifting fitness function, this method employs several algorithmic adaptations from the aforementioned multi-objective optimization. Two notable changes have been implemented. First, the input case pairs needed to calculate GATEness may not be present in the three randomly selected input case pairs. Consequently, this method’s relevant power values at a given driving frequency are amalgamated into a fitness metric for ω_1 and ω_2 as the product of three fitness terms for ω_1 and ω_2, respectively. Specifically, if a given input case should result in a ‘0’ output then the fitness term is $\frac{1}{\hat{f}_{\omega_n}}$, while if a given input case is that should results in a ‘1’, then the fitness term is $\frac{\hat{f}_{\omega_n}}{1}$. Similar to the GATEness metrics, this method rewards high fitness for materials that minimize power for any output that should be a ‘0’ output and maximize power for any output that should be ‘1’. Another algorithmic adaptation required is that when a new set of three input case pairs gets selected at each generation, all individuals in the population must be re-evaluated by the latest fitness function. This is because the chosen input case pairs directly influence the magnitude of fitness. Without re-evaluation, there would be stagnation and evolutionary progress would not proceed. The number of simulations this method requires scales as $O(1)$ with the n the number of gates embedded within a material.

Tri-Objective. The final evaluation method we test follows the same process as the simultaneous method while introducing a third objective for independence on a subset of three input case pairs. By adding an objective for independence in a subset of input case pairs, we explicitly evolve for independence without having to perform $O(2^n)$ operations for each material. To construct the independence objective, we simulate material behavior in 3 randomly selected input case pairs. To ensure we can create a binarizing threshold by averaging the power for the lowest value of ‘1’ and the highest value of ‘0’, we enforce that at least one of the selected input case pairs results in a logical output of ‘1’, and one results in a logical output of ‘0’. At every generation, we select a new subset of input cases on which to evaluate independence. The independence is then a value $\in$ [0,3]. The number of simulations this method requires scales as $O(1)$ with the n the number of gates embedded within a material.

2.7 Survivor Selection

After all the materials are evaluated, those allowed to maintain in the population are either selected based on the Pareto-front of fitness_{g_1} and fitness_{g_2} or, in experiments using the tri-objective evaluation method, off of the Pareto-front of fitness_{g_1} and fitness_{g_2} and independence. Survivor selection occurs by iteratively selecting two random individuals from our population, and discarding one if it

is Pareto-dominated. This process repeats until the population size reaches λ. Each of the remaining materials is allowed to produce a mutated copy of itself until the population size reaches $(\lambda + \mu)$.

2.8 Mutation

Our mutation operator acts on each child with a 10% chance of mutating the mass of a given grain in the child's genome. The mutation size is $\in \pm 0.0524\,\text{kg}$. Grains are limited to a minimum mass of $0.262\,\text{kg}$ and a maximum mass of $5\,\text{kg}$. We repeat this process of creating, evaluating, and mutating materials until 100 generations have occurred.

3 Results

After evolution, we analyze each individual in the population at generation 100. We retrieve the material with the highest independence I from the population. This is repeated for all 30 evolutionary replicates. Table 1 shows the mean independence of these 30 top-performing materials in each evaluation method.

Table 1. Comparison of our four evaluation methods and random. The mean independence of materials evolved using our distributed, simultaneous, input case varying, and tri-objective evaluation methods. Here, big-O refers to how the number of physics simulations scales with n the number of logic gates embedded within a material.

Evaluation Method	Gate 1	Gate 2	Mean Independence	Big-O
Distributed	AND	AND	12.30 ± 1.77	O(n)
Simultaneous	AND	AND	12.60 ± 2.04	O(1)
Input Case Varying	AND	AND	10.80 ± 2.45	O(1)
Tri-Objective	AND	AND	12.86 ± 1.83	O(1)
Distributed	XOR	AND	14.70 ± 1.00	O(n)
Simultaneous	XOR	AND	14.03 ± 1.62	O(1)
Input Case Varying	XOR	AND	14.93 ± 0.35	O(1)
Tri-Objective	XOR	AND	14.46 ± 1.28	O(1)

To assess whether evolution produced materials with greater independence than randomly generated ones, we created a population of 1000 random materials and evaluated the fitness of each material. For all evaluation methods, evolved materials had significantly higher independence than the randomly generated ones ($p < 0.05$ for all comparisons).

AND & AND. When the logic gates g_1 and g_2 embedded within a material are logical AND, the tri-objective evaluation method produced the materials with the highest mean independence. In this method, 11 of the 30 replicates produced materials with perfect independence $I = 15$. The method that produced the second highest mean independence in the simultaneous method, followed by the distributed method. Respectively, using these methods, 10 and 6 replicates produced materials with perfect independence $I = 15$. The worst-performing evaluation method was the input case varying method. It resulted in materials with the lowest mean independence and only 5 replicates produced materials with perfect independence. Using the Mann-Whitney U test, we conclude that the independence of materials from the input case varying evaluation method is significantly less than material evolved using the other three methods ($p < 0.05$ for all comparisons). There is no significant difference between the distributed, simultaneous, and tri-objective methods ($p > 0.05$ for all comparisons). Figure 4 shows the behavior of sample $I = 15$ materials from each of the evaluation methods.

XOR & AND. When logic gate g_1 is XOR and logic gate g_2 is AND, the input case varying evaluation method produced the materials with the highest mean independence. Using this evaluation method, 29 of the 30 replicates produced materials with perfect $I = 15$ independence. The method that produced the second highest population mean fitness is the distributed method, which produced $I = 15$ materials in 27 of the 30 replicates. The tri-objective evaluation method produced the third highest population independence and found $I = 15$ materials in 25 of the 30 replicates. The worst-performing evaluation method was the simultaneous one, which produced the lowest mean population independence and found $I = 15$ materials in 22 of the 30 replicates. There is no significant difference between the independence of materials evolved from any of the methods.

4 Discussion

In this work, we have shown that it is possible to evolve entirely independent computational granular materials without explicitly selecting for independence. Our results unveil computationally efficient methods of embedding logic gates into granular metamaterials. We proceed with analyzing our findings.

4.1 Relative Independence of AND & AND Vs XOR & AND

Comparing within each valuation method, we find that materials where g_1 = AND, g_2 = AND have significantly lower independence than materials where g_1 = XOR, g_2 = AND ($p < 0.05$ for all comparisons). That is, materials with heterogeneous logic gates display higher independence in all evaluated methods tested. It is not clear why this is the case. One hypothesis is that in materials

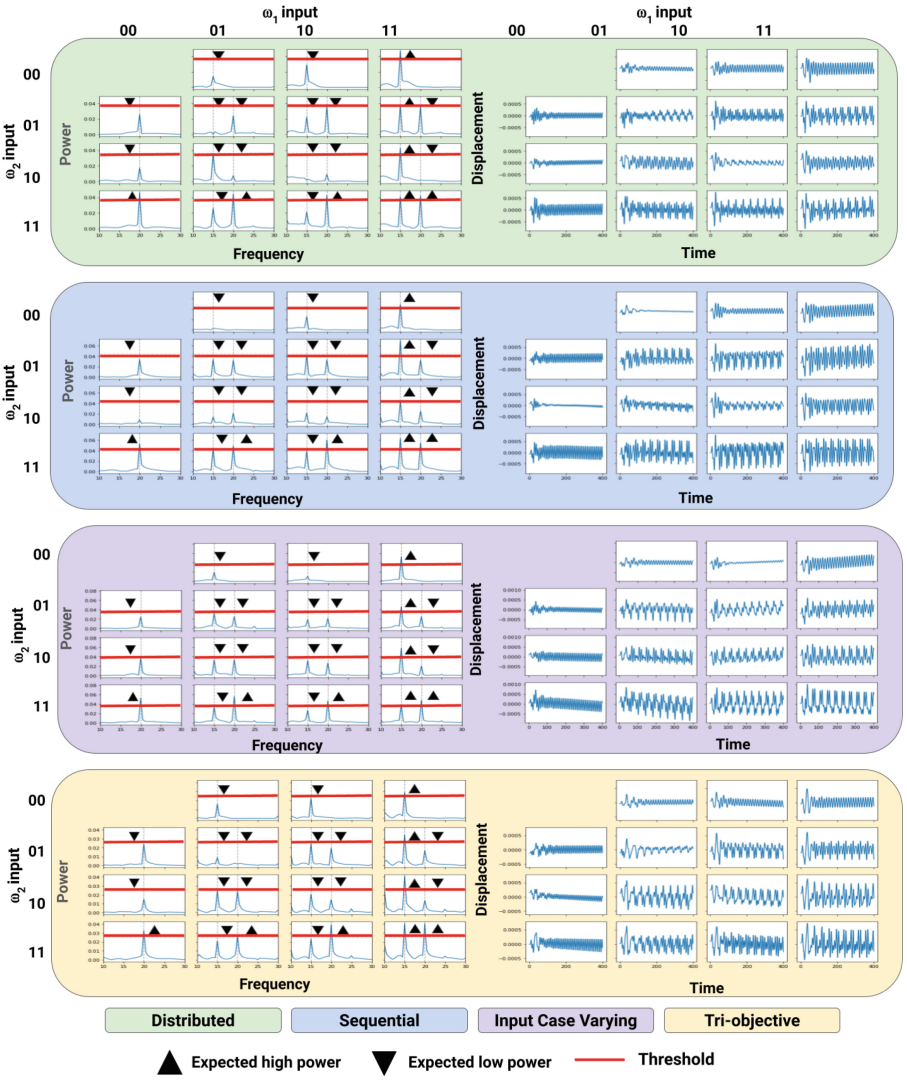

Fig. 4. Comparison of sample material behavior where $g_1 = \mathbf{AND}$, $g_2 = \mathbf{AND}$. This figure compares sample behavior for materials evolved across our four evaluation methods. The panels on the left side show behavior in the frequency domain, while the panels on the right side show behavior in the time domain. Each material here has a perfect independence score of $I = 15$.

with homogeneous logic gates, each gate might rely on the same physical phenomena to complete its logical function. For example, each logic gate could rely on constructive interference between the same pair of grains at a certain location within the material. Therefore, when the gates are supplied with different

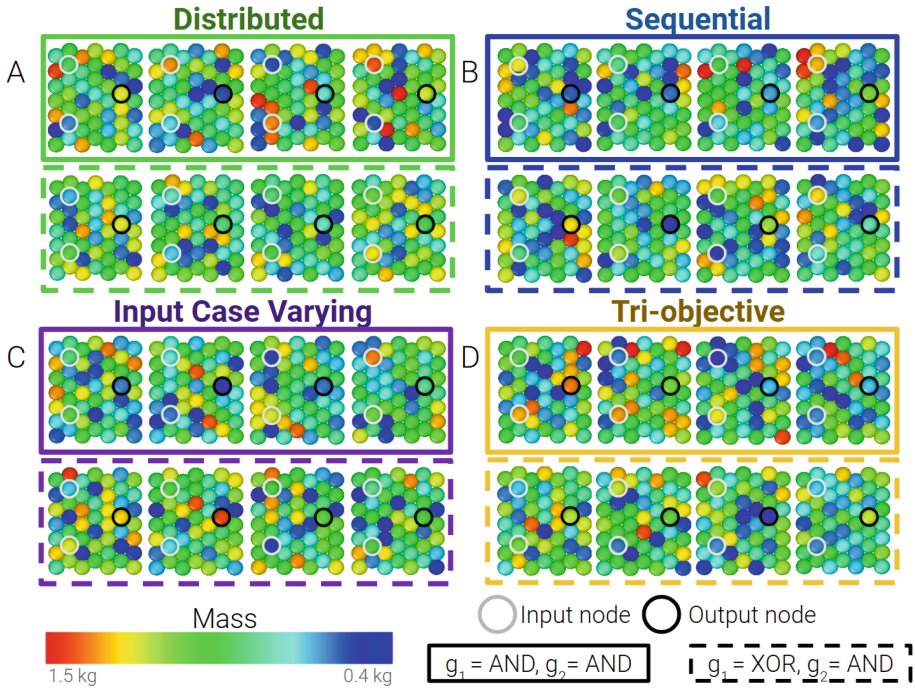

Fig. 5. Comparison of material configurations. This figure displays configurations from the four $I = 15$ materials from each of our evaluation methods both where g_1 = AND, g_2 = AND (materials with solid outline), and where g_1 = XOR, g_2 = AND (materials with dashed outline).

input cases, the internal pathways on which they each rely to complete their function are polluted by interference from the other gate. Further investigation is required to understand the relationship between logic gate choice and logical independence. Further investigation is also required to understand if this behavior is present for other logic gate choices, such as where g_1 = XOR and g_2 = XOR or g_1 = AND, g_2 = NAND. If this finding is ubiquitous across other logic gate pairings, it would have implications in the application of CGMMs. Specifically, it would guide CGMM design and implementation such that we would avoid embedding homogeneous logic gates into a CGMM, in preference for CGMMs with mixed logic gates.

4.2 Evolved Material Configurations

It is interesting to visually compare the configurations of materials both between and within each evaluation method. Figure 5 displays the configurations of four materials with $I = 15$ independence materials that were evolved using each of the tested evaluation methods, both where g_1 = AND, g_2 = AND, and where

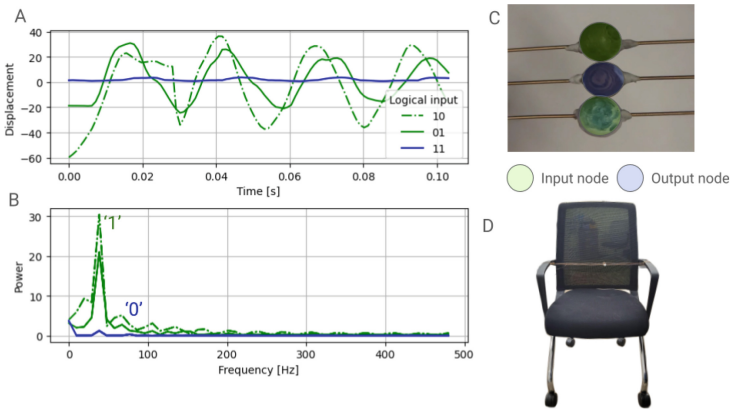

Fig. 6. Physical implementation of a granular logic gate. A) The time domain displacement of the input grain under each logical input. **B)** The frequency domain signal under each logical input. Here, we expect a high power at the driving frequency for each of the green lines, and a low power at the driving frequency of the blue line. The observed behavior here is consistent with the expected behavior of an XOR logic gate. **C)** This image shows a close-up of our 3-bead system. **D)** This photograph displays our entire chair-XOR system.

g_1 = XOR, g_2 = AND. We encourage the reader to closely examine the configurations of these materials. Interestingly, each of the materials depicted appears to have distinct grain-mass organizations. There are very few discernible patterns among the materials. This includes no discernible pattern in the properties of input grains, output grains, or border grains. This applies to both materials where here g_1 = AND, g_2 = AND, and where g_1 = XOR, g_2 = AND. The absence of clear patterns in material properties underscores the intricate and unintuitive relationship between form and function in CGMMs. This diversity of configuration reveals a breadth of functional material arrangements that can arise within the design space.

5 Hardware Implementation

To further the physical realizability of CGMMs, we proceed with implementing a simple physical computational granular XOR. We use everyday materials, including an office chair, guitar strings, adhesive putty, and beads. Our implementation consists of three grains strung onto parallel guitar strings placed across the armrests of an office chair. Figure 6C shows a close-up view of the system, while Fig. 6D shows a photo of the entire chair system. The two outer beads act as input grains and the center bead acts as an output grain. The grains' y-locations were adjusted such that the beads were in light contact when at rest. Their locations in the x-plane were aligned and then maintained by adhesive putty. In this system, an input of '1' is passed into the top grain as a hand-pluck

with amplitude $\approx$ 2.5 cm applied 90°C to the string, while an input of '1' is passed into the bottom grain as a hand-pluck with amplitude $\approx$ 2.5 cm applied 270°C to the guitar string. This system is designed such that there is destructive interference from the input grains when the '11' input case is passed into the system, leading to little movement of the output grain.

To assess the function of this system, we recorded its behavior under the '01', '10', and '11' input cases. The '00' input case is not considered as it is trivial. The behavior of the output grain was captured on video with a frame rate of 960fps using Super Slo-mo mode on a Galaxy S21 Android phone. Each video was then processed to extract the displacement of the output grain under each logical input using trackR [6]. These videos are included in the supplementary materials. Figure 6A and 6B respectively show the behavior of the output grain over time, and across frequency. Both of these figures support that our system exhibits appropriate XOR behavior.

6 Conclusions & Future Work

This study presents scalable methods for evolving independent polycomputational granular materials without explicitly optimizing for independence. We report the first discovery of entirely logically independent CGMMs. Notably, we observed that evolution discovers more independent CGMMs when a material has mixed logic gates compared to those with homogeneous gates. However, the extent to which this trend generalizes to other gate pairings remains unclear. Resolving this uncertainty will be a focus of future investigations. We also implement a vibrational XOR gate in hardware. While not AI-designed, this gate serves as a physical proof-of-concept for computational functionality in vibrational granular assemblies. Expanding this work to include more complex, two-dimensional grain structures will be the next step to transferring evolved designs from simulation to reality.

Acknowledgements. We want to thank Frederic Sansoz for his guidance with LAMMPS and Emily Ertle for her assistance with the hardware implementation. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. 2235204. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. We would like to acknowledge financial support from the National Science Foundation under the DMREF program (award number: 2118810).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Beaulieu, S., Welch, P., Parsa, A., O'Hern, C., Kramer-Bottiglio, R., Bongard, J.: Refractive computation: parallelizing logic gates across driving frequencies in a mechanical polycomputer. In: ALIFE 2024: Proceedings of the 2024 Artificial Life Conference. MIT Press (2024)
2. Bongard, J.: From rigid to soft to biological robots: how new materials are driving advances in the study of the embodied cognition. Artif. Life Robot., 1–5 (2023)
3. Bongard, J., Levin, M.: There's plenty of room right here: biological systems as evolved, overloaded, multi-scale machines. Biomimetics **8**(1), 110 (2023)
4. Córcoles, A.D., et al.: Challenges and opportunities of near-term quantum computing systems. Proc. IEEE **108**(8), 1338–1352 (2019)
5. van De Burgt, Y., Melianas, A., Keene, S.T., Malliaras, G., Salleo, A.: Organic electronics for neuromorphic computing. Nature Electron. **1**(7), 386–397 (2018)
6. Garnier, S.: trackR - Multi-object tracking with R (2022). https://swarm-lab.github.io/trackR/, r package version 0.5.3
7. Gorecki, J., et al.: Chemical computing with reaction-diffusion processes. Philos. Trans. Royal Soc. A: Math. Phys. Eng. Sci. **373**(2046), 20140219 (2015)
8. Hameed, S.A.: Controlling computers and electronics waste: toward solving environmental problems. In: 2012 International Conference on Computer and Communication Engineering (ICCCE), pp. 972–977. IEEE (2012)
9. Horowitz, M., Grumbling, E.: Quantum computing: progress and prospects (2019)
10. Kim, E., Yang, J.: Wave propagation in granular metamaterials. Funct. Compos. Struct. **1**(1), 012002 (2019)
11. Knill, E.: Scalable quantum computing in the presence of large detected-error rates. Phys. Rev. A **71**(4), 042322 (2005)
12. Parsa, A., Wang, D., O'Hern, C.S., Shattuck, M.D., Kramer-Bottiglio, R., Bongard, J.: Evolving programmable computational metamaterials. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 122–129 (2022)
13. Parsa, A., Wang, D., O'Hern, C.S., Shattuck, M.D., Kramer-Bottiglio, R., Bongard, J.: Evolution of acoustic logic gates in granular metamaterials. In: International Conference on the Applications of Evolutionary Computation (Part of EvoStar), pp. 93–109. Springer (April 2022)
14. Parsa, A., Witthaus, S., Pashine, N., O'Hern, C., Kramer-Bottiglio, R., Bongard, J.: Universal mechanical polycomputation in granular matter. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 193–201 (2023)
15. Paun, G., Rozenberg, G., Salomaa, A.: DNA computing: new computing paradigms. Springer Science & Business Media (2005)
16. Tanaka, G., et al.: Recent advances in physical reservoir computing: a review. Neural Netw. **115**, 100–123 (2019)
17. Theis, T.N., Wong, H.: The end of Moore's law: a new beginning for information technology. Comput. Sci. Eng. **19**(2), 41–50 (2017)
18. Thompson, A.P., et al.: LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. Comp. Phys. Comm. **271**, 108171 (2022). https://doi.org/10.1016/j.cpc.2021.108171
19. Vega, A., Bose, P., Buyuktosunoglu, A.: Rugged embedded systems: computing in harsh environments. Morgan Kaufmann (2016)
20. Xu, X.Y., Jin, X.M.: Integrated photonic computing beyond the von Neumann architecture. ACS Photonics **10**(4), 1027–1036 (2023)
21. Yasuda, H., et al.: Mechanical computing. Nature **598**(7879), 39–48 (2021)