1. Introduction

The fields of bioinspired robotics and soft robotics center on the ideas of self-organization and embodiment, which capitalize on the reciprocal and dynamic coupling among the brain, body, and environment.\(^1\) One of the pioneering works that illustrates the importance of this concept is the passive dynamic walker by McGeer.\(^5\) This robot walks down a slope using self-stabilization that occurs in the mechanical interaction between its body and the slope, without control by an electrical circuit or actuation, and its behavioral control appears to be as natural as a human walker. Since the creation of the passive dynamic walker, many intriguing robots and techniques have been developed in the field of locomotion and walking behavior control.\(^6,7\) Soft structures enhance the dynamic coupling between the body and environment and are expected to help increase adaptability and generate additional functionality. Shepherd et al. proposed a pneumatically driven silicone-based soft robot that can implement multiple gait patterns in undulatory locomotion, where these diverse patterns have emerged as an outcome of body–environment couplings.\(^8\) Brown et al. proposed a jamming-based gripper for picking up various objects without electrical feedback.\(^9\) The contact interaction between the object and the gripper induces deformation in the shape of the gripper and autonomously adapts to the morphology of the grasped object. In these studies, control was partially outsourced to the morphological and material properties. When morphology and materials perform functions that are usually attributed to the brain or control, the phenomenon is called “morphological computation.”\(^10\) Recently, numerous attempts have been made to implement computational architectures in the mechanical realm.\(^11\)–\(^14\)

Physical reservoir computing (PRC) exemplifies one recent attempt that specifically exploits the natural dynamics of the physical body as a computational resource.\(^15\)–\(^17\) Nakajima et al. demonstrated that the physical dynamics of a soft robotic arm inspired by an octopus can be used for implementing closed-loop control\(^18\) and for emulating complex nonlinear dynamical systems,\(^19,20\) tasks that are conventionally implemented using a recurrent neural network running on a PC. Tanaka et al.
demonstrated a method for identifying the direction of a wind stream from the movement of a soft flapping wing measured by flexible strain sensors as PRC. An increasing number of PRC applications has been recently proposed for many different soft robotic platforms, such as tensegrity structures, pneumatically driven systems, fish robots, origami robots, and so on. In these examples, because the computational load is mainly outsourced to the physical body, we can expect to see savings in information processing. Furthermore, PRC can deal with memory storage and its processing simultaneously in terms of dynamics, which differs completely from the conventional von Neumann architecture, wherein the memory and central processing units are separate. This frees the system from its intrinsic processing speed limitations (often termed “von Neumann bottleneck”) and makes PRC particularly well suited for edge computing devices, where quick processing of the obtained data is crucial.

This study aims to further extend the application domain of PRC and make the framework radically more flexible for soft robotic applications. Instead of using reservoirs, whose physical constituents are predefined and fixed, we aim to introduce the novel concept of “self-organized remote reservoirs,” which can autonomously alter their physical constituents in real time. These remote reservoirs will be generated in a self-organized manner, such that they can function as successful reservoirs by including many ad hoc physical substrates in the environment; thus, they are capable of transferring computation spatially distant places. This extension was made possible because of two insights. The first insight comes straight from the mechanism of reservoir computing. Although the reservoir computing framework allows us to exploit complex, high-dimensional dynamics as a computational resource, not all types of dynamics can be effectively and reliably used. Certainly, there exist prerequisites for a successful reservoir and a key condition is an “echo state property” (ESP), which requires a reservoir state to be a function of the input stream only. This property guarantees a reproducible response against the input stream because the reservoir state does not depend on the initial condition of the system but instead depends only on the input stream, which is essential for implementing reliable computation. This situation suggests that the input-driven dynamics of the reservoir should synchronize with the input stream, whose phenomena are often termed “common-signal-induced synchronization” or “generalized synchronization” (note the difference from complete synchronization). Subsequently, the second insight arises. If generalized synchronization between the input stream and reservoir state guarantees a successful reservoir, can we use physical dynamics that synchronize with the input stream through body–environment couplings? This insight is also a direct consequence of the concept of embodiment.

This article’s goal is to demonstrate whether a self-organized remote reservoir works using a simple physical platform through systematic experimentation. We prepared two silicone rubber strips immersed in a water environment, where one rubber strip received direct actuation from the motor and the other was passively located in a distant place from the actuated rubber strip (Figure 1). We also developed a mechanism for embedding a strain sensor inside the silicone rubber strips using carbon nanotubes (CNTs) and liquid metal to retain the softness of the material and its sensitivity to environmental stimuli. Because the two soft silicone rubber strips were coupled through the water medium, it was expected that the motion of the actuated silicone rubber strip would affect the motion of the passive one and vice versa. How, then, does the modality of the interaction between the two silicone rubber strips change based on the distance between the two? Does the passive silicone rubber strip synchronize with the input actuation commands? If yes, does this contribute to transferring computation from one silicone rubber strip to the other, and further, which type and amount of computational power will be transferred? We aim to clarify these questions in detail. A similar study focusing on the effect of a water medium was previously proposed by Judd et al. They demonstrated a method for estimating the position of an underwater object from the movement of a soft robotic arm in the same water environment without directly touching or seeing it. In this case, the position of the object affected the movement of the arm through the water medium, resulting in indirect sensing and position detection using the dynamics of the arm as a reservoir. Our approach extends this perspective and aims not only to modulate the dynamics of the arm, but also transfer effective computation to a distant place using the water medium.

This article is organized as follows. In the next section, our experimental setups are explained, including the information processing scheme that is based on reservoir computing. Next, the dynamics and interaction modalities of remote reservoirs are analyzed. Subsequently, several benchmark tasks are implemented to investigate whether they can be used for information processing. The expressiveness of remote reservoirs is also revealed in terms of information processing capacity (IPC), exploring how the capacity is modulated according to the settings of body–environment couplings. Finally, Discussion illustrates the potential applications of our approach.
2. Results

2.1. Experimental Setup

2.1.1. Physical Platform

Figure 1 shows the overall physical platform of the experiment. Two silicone rubber strips were immersed in the water tank. Each silicone rubber strip was equipped with flexible strain sensors embedded in its body (Figure 2). The length and thickness of the silicone rubber strips were 14 cm and 2 mm, respectively. A servo motor (Dynamixel MX-64) moves one silicone rubber strip (called the “Active silicone rubber strip”), and this affects the other silicone rubber strip (called the “Passive silicone rubber strip”) through direct contact or through the medium of water. The motion of the silicone rubber strips was captured from the side at 30 fps, and the outputs of the strain sensors were recorded at 1000 Hz.

The motor received the command input every τ timesteps (set as τ = 300 ms) and controlled the joint angle. The command input sequence was prepared as a series of independent and identically distributed (i.i.d.) binary random variables. The binary values were the two target positions, which were 45° from the vertical downward and 90° apart. The series of 5000 command inputs consisted of 0 and 1 values, each with a probability of 0.5. The outputs of the strain sensors were sampled at 1000 Hz.

Figure 2. The flexible strain sensor used in this article. A) Schematics of the flexible strain sensor. B) Scanning electron microscopy image of the CNT network in the flexible strain sensor. C) Strain sensing mechanism. The red and yellow dots highlight the electrical connection points between CNTs, and yellow dots describe the disconnection under strain. Bending imposes a tensile strain (ΔL). D) Normalized resistance change ratio as a function of the bending radius. E) Real-time sensor response when the sensor structure is waved. Inset photos show (left) zero strain and (right) under strain.
inputs was sent to the system in one experimental session; this took 1500 s. Three series of commands were tested.

The behavior of the two silicone rubber strips changed as the distance between them (termed "distance") increased. One rubber strip contacted with another rubber strip when the distance was small. Thus, the distance between the two silicone rubber strips was changed to 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, and 25 cm. The same command input sequence was attempted twice for each distance.

A sensory response $s$ was obtained from a single strain sensor in real time for each silicone rubber strip. The sensory response in the silicone rubber strip that was directly actuated by the motor was called “Active.” The sensory response in another silicone rubber strip that moved passively according to the environmental change was called “Passive.” The value of the control input was called “Input.”

2.1.2. Preparation of the Reservoir States

Our aim was to exploit the physical dynamics of the silicone rubber strips as a computational resource, specifically as a reservoir. As an expression of physical body dynamics, a sensory response from the flexible strain sensor was employed for each silicone rubber strip. To prepare the reservoir states from the sensory response of the silicone rubber strips, a technique called time-multiplexing was used. This technique exploits the time scale difference between the input command series and response series, preparing temporally spaced computation nodes; this technique is common in the PRC field. Several applications of this technique can also be found in PRC using soft robots.[20,21,39] The time series of the sensor $\{s_k\}$ was multiplexed to prepare the reservoir states $x_n$ to be a response to the $n$th input command $u_n$ as follows (Figure 3).

$$x_n = \left[ x_{n,1}, x_{n,2}, \ldots, x_{n,60} \right]$$

$$x_{n,k} = \frac{1}{5} \sum_{l=0}^{4} s_{n+k+1}$$

where $x_{n,k}$ is the mean of $s_t$ (note that $t = \tau n + l$) in the $k$th time interval (a single interval consists of five timesteps throughout this article) of the 60 intervals when the $n$th input command was executed.

The reservoir states were prepared in the same manner for each silicone rubber strip. When we used both Active and Passive silicone rubber strips as a reservoir (termed "Both" or "Both reservoirs"), we simply concatenated both reservoir states and considered them as a single large reservoir. This approach is called spatial multiplexing and its performance has been well-studied, showing that any combination of subreservoirs in the spatial multiplexing always improves the performance if there is no overfitting or the state matrix is full rank.[40] In our experiment, to implement a Boolean function emulation task, a simple logistic regression was applied to train the readout, taking the reservoir states as its inputs and the outputs as a required binary state. The setting of the training procedure is given later in more detail. The IPC was also calculated based on the reservoir states.

![Figure 3. Preparing the reservoir states from the sensory time series by time multiplexing. Sensory time series from both the Active and Passive silicones are used. A single input takes a time length of $\tau$, which is set at 300 timesteps throughout the article, the corresponding sensory time series are averaged for every five timesteps, and the resulting 60 locally averaged values are used as reservoir states for each type of silicone. In the experiments, three combinations of the reservoir states (Active, Passive, and Both) are tested.](image-url)
2.1.3. The Flexible Strain Sensor

The strain sensor, which detects the movement of the silicone rubber strip, is briefly explained. Figure 2A displays a schematic and photo of the strain sensor. To convert strain into an electrical signal, a single-walled CNT film was printed onto a silicone rubber strip as a strain-sensing material using a dispenser. For the interconnection, liquid metal GaInSn (galinstan) was also printed because it is stretchable and highly conductive material compared with the CNT film. This resulted in stable measurements under the movement of the silicone rubber strip structure. To make the electrical signal output to the measurement system, Ag electrode-printed polyethylene terephthalate (PET) film was laminated on the silicone rubber with galinstan. Finally, to make a waterproof strain sensor, a silicone rubber strip cap was used to cover the sensor and PET film.

Next, the sensing mechanism and performance of the strain sensor are discussed. When the silicone rubber strip moves, strain is applied to the CNT film. Because of the fully packed CNT network film (Figure 2B), when only tensile strain is applied, electrical resistance increases because the CNT connection points decrease, as described in Figure 2C. To confirm this behavior, electrical resistance was measured by varying the bending radius of the strain sensor (Figure 2D). The result shows that resistance increases when the bending radius decreases (i.e., higher strain). To shed more light on sensor performance, when force was applied to the silicone rubber strip at the beginning, the swing movement was measured using the strain sensor. At first, the film pendulum had a wide swing, but gradually the motion reduced, corresponding to reduced strain reported by the sensor. This behavior was clearly observed by the strain sensor, as shown in Figure 2E. It should be noted that sensor output at near-zero strain gradually decreased the film pendulum cycles, as shown in Figure 2E. This is because the response time of the sensor was slower than the pendulum speed. To address this issue, sensor performance needs to be improved to detect response to the strain more quickly.

2.1.4. Experimental Procedures

The experiments were conducted and the measured data were analyzed according to the following procedures. First, to investigate the difference in the responses of the sensors (i.e., Active and Passive), their behavior was observed and plotted. Second, to investigate the relationships between Input, Active, and Passive, the cross correlations were calculated. Third, to investigate whether the rubber strips and water displayed the characteristics of a physical reservoir, ESP was investigated by analyzing the behavior of the reservoir states against the same input sequence and then different input sequences and subsequently comparing the two results. Fourth, to investigate whether a remote object can be used for computation, Boolean function emulation tasks were implemented using the reservoirs prepared from Active, Passive, and Both to analyze the performance in each condition in detail. Fifth, to investigate how the information-processing capability is modulated according to the interaction modality between the silicone rubber strips and the water medium, the IPC, which can reveal the nonlinear memory capacity of the dynamics, was analyzed in detail.

2.2. Experimental Results

2.2.1. Response of the Strain Sensors

Figure 4 shows the command input, the angle, Active, and Passive when the same command inputs were given continuously (Figure 4A, step response) and when the target dynamically changed (Figure 4B, random command). Figure 4A shows that the time in which the corresponding response pattern appears in Passive increases as the distance increases, while the time in Active does not change. This behavior is consistent when actuating the system with random commands (Figure 4B). These results indicate that it takes time for the Passive silicone rubber strip to be affected by the command change (see Supplementary video).

2.2.2. Analyzing the Interaction Modality between Silicone Rubber Strips

Figure 5A shows the behavioral change in the silicone rubber strips. The two silicone rubber strips moved together, sticking to each other in conditions where the distance between the strips was 2, 3, 4, 5, 6, and 7 cm (Figure 5A top left). The end of one strip touched another in conditions where the distances were 8, 9, and 10 cm (Figure 5A, top right). The strips did not touch each other in conditions where the distances were 15, 20, and 25 cm (Figure 5A, bottom). The number of contacts between the strips were measured during each run, and the results were 891, 926, 994, 1006, 717, 327, 330, 113, 60, 0, 0, and 0 times when the distance was 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, and 25 cm, respectively. The reason for the growth of the number of contacts in concert with the increase in distance in the range of 2–5 cm is that the two strips first stick together and then start to peel off as movement continues.

Figure 5B shows the results of the cross correlations. The top of Figure 5B indicates that the correlation profiles between Input and Active do not change by distance. The center of Figure 5B indicates that the delay value, in the correlations between Input and Passive peaks, increases as the distance increases. Furthermore, when the distance is short, the patterns of correlation behave similarly to those between Input and Active. This outcome can be seen by observing the behavior of the strips that directly contact and stick to each other (see Supplementary video). The bottom of Figure 5B shows the same tendency as the center of Figure 5B in the correlations between Active and Passive. These results also indicate that it takes time for the effect of the command change to reach the Passive silicone rubber strip, as shown in Figure 4.

The behavioral changes correspond to the cross correlations. In conditions where the two silicone rubber strips moved together and stuck to each other (Figure 5A, top left), the correlations between Input and Active and between Input and Passive show the same changes (Figure 5B, top and middle). In contrast, in conditions where the strips did not touch each other (Figure 5A, bottom), the correlations between Input and Passive show the same changes (Figure 5B, bottom).
Passive with an 800–1500 ms delay are larger than the ones between Input and Active (Figure 5B, top and middle).

2.2.3. Investigating the Echo State Property

When a dynamical system is given the same input sequence and converges to the same state after starting from different initial states, this system can be considered to have an ESP,[32,33] suggesting that it can act as a successful reservoir. Thus, we investigated whether the response of the silicone rubber strips has an ESP. The distance between $x_n$ and $\tilde{x}_n$ was calculated as $|x_n - \tilde{x}_n|$, where $x_n$ and $\tilde{x}_n$ are the reservoir states generated from the same input sequence starting from different initial states. The distances for both cases, when the same input sequence was used and...
that when the different input sequences were used, were compared. A small distance implies that the system has an ESP and has converged to a similar state, even if the system started with different initial states and was driven by the same input sequence.

Figure 6A shows the time average for each distance $|x_n - \bar{x}_n|$. The distance of Active and Passive equals 2.0 when different input sequences were applied, while the distances were much smaller when the same input sequence was applied. Considering that these states are from the physical experiment and include noise in the sensory values, the figure indicates that both Active and Passive reservoir states have a tendency to show generalized synchronization with the input stream and, therefore, a tendency to show an ESP.

2.2.4. Implementing Boolean Function Emulation Tasks

Here, to evaluate whether the proposed remote reservoir actually works, Boolean function emulation tasks were implemented using the reservoir states of the two silicone rubber strips. Based on the PRC framework, the readouts attached to the silicone rubber strip were trained to emulate the parity checker and a function that reproduces previous inputs, which is often called a short-term memory task.\[18,41,42\] The target output of the N-bit parity checker (termed N “input case”) at input timestep $n$ is $y_n = q(\sum_{d=1}^{N} u_{n-d})$, where $q(x) = 0$ if $x \equiv 0$(mod$2)$ else $q(x) = 1$ and $u_n$ is the input at input timestep $n$. The target output of the short-term memory task at the $n$th input timestep is $y_n = u_{n-N}$ (termed N “delay case”). Note that the short-term memory task only requires linear memory but that the parity checkers require not only memory but also nonlinear processing to implement.

Readouts were trained using logistic regression from the time series of time-multiplexed data and the target output of the computational tasks. The first 1000 steps were removed and 5000 steps of data were used—300 for washout, 3700 for learning, and 1000 for evaluation. The readouts were trained and tested for each distance, each sensor pattern (Active, Passive, Both), and each trial, and the correct response rate (the rate of correct responses in evaluation) was calculated. This rate is termed “capacity” in our analysis, where the summation of the square of the correlation between the target and system output is calculated over delays (up to 25 timesteps before).
Figure 6B shows the behavior of the target and system output of a 2-bit parity checker emulation task (see Supplementary video, Supporting Information). It was confirmed that the passive reservoir, which is a remote reservoir, can perform the task. Furthermore, the performance improves using Both reservoirs, suggesting the positive contribution of the remote reservoir.

Figure 6C shows the capacity of the parity checker and short-term memory task. First, we focused on the capacity results of the parity checker tasks. In these tasks, it can be confirmed that the passive reservoir, which is a remote reservoir, can perform the task. Furthermore, the performance improves using Both reservoirs, suggesting the positive contribution of the remote reservoir.

Figure 6.
Analyzing the reservoir dynamics and its Boolean function emulation task performance. A) Investigation of the ESP of reservoir dynamics in the Active and Passive silicone rubber strips. The time average of the distance between the two reservoir trajectories, $|x_n - \bar{x}_n|$, as driven by the same input sequence, is plotted for each condition of distance. The error bars show the standard deviations. As a comparison, the time average of the distance between the two reservoir trajectories driven by different input sequences is also plotted. B) Typical output time series for a 2-bit parity checker emulation task. The cases are plotted from top to bottom, using both the Active and Passive silicone rubber strips, only the Active silicone rubber strip, and only the Passive silicone rubber strip. The target time series is also aligned for each plot. C) Task performances analyzed through the averaged capacities for each distance and choice of silicone rubber strip (both Active and Passive, only Active, or only Passive). The shaded region shows the standard deviation for each plot. Each diagram shows the results for a different task. The upper line shows parity-checker task cases, and the lower line shows the short-term memory task cases.
within a certain distance (around 8 cm), the Passive reservoir can perform the task, but at distances of 9 cm and longer, its performance was unsuccessful. Within this range, both reservoirs show better performance than using either reservoir individually (such as in the 3 Input case, with a distance of around 4 cm). However, it should be noted that the performance of the Active reservoir degrades a little within this range (degradation is most significant with a distance of 8 cm). We speculate that this is because the silicone rubber strips interact through direct contact at this range and the Passive strip might act as a perturbation to the Active strip.

In short-term memory tasks (such as the 1-delay case) that require shorter delays, the behavior of the capacities shows a similar tendency to the parity checker case (see also Supplementary video, Supporting Information). However, interestingly, when the task requires a longer delay, the capacity of the Passive reservoir grows, even exceeding the capacity of the Active reservoir, especially when the distance is large (such as, the distance larger than 7 cm in the 3-delay case). This may be caused by the delayed response of the Passive silicone rubber strip shown in Figure 4. Here, a performance improvement can also be found using both reservoirs, which show better performance than using either of the reservoirs (such as in the 2-delay case with a distance of around 4 cm). These results indicate that using the Passive reservoir leverages the computational tasks. That is, a remote object can be used for computation in PRC.

2.2.5. Analyzing the Expressiveness of Physical Reservoirs Using Boolean Capacities

The expressiveness of our reservoirs was clarified using IPC. IPC indicates the capacity of a dynamical system to approximate some products of orthogonal polynomial functions (see the Experimental Section for details). Because the capacity was defined for the binary input stream in our experiment, this IPC was called Boolean capacity. In the following analysis, 1d capacity indicates a sum of linear memory capacities over delay (up to 25 timesteps before) and nd capacity indicates a sum of nonlinear memory capacities with the nth degree (up to 25 time-steps before).

Figure 7A indicates that Passive and Both have 1d capacity with a long delay in cases of a large distance, while Active does not have 1d capacity with a large delay. This result is consistent with our observations in the previous sections. Figure 7B shows that the capacity of Both was larger than Active, and Passive had larger capacity when the distance was 3, 4, and 5 cm than Active. Figure 7B also indicates that Passive and Both have 4d and 5d capacity while Active does not. These results indicate that Passive reservoirs are particularly suited for nonlinear computation when the distance is short, but they have long linear memory when the distance is long; interestingly, in some of these cases, the total amount of IPC for the Passive reservoir was larger than that of the Active reservoir. Furthermore, we can clearly confirm that the Passive reservoir can enhance the overall capacity when combined with the Active reservoir.

3. Discussion

In this study, we have shown that a simple experimental setup using two silicone rubber strips embedded with flexible strain sensors is capable of transferring computation to a spatially distant place, namely, from the actuation point to the Passive silicone rubber strip. This is realized by generalized synchronization through effective coupling between the soft silicone rubber strips and a water medium, simultaneously implying that the passive silicone rubber strip fulfills the ESP and successfully acts as a remote reservoir. The “transferring” of computation occurs through the physical coupling including delay to propagate the input information from Active to Passive silicone rubber strips.

**Figure 7.** Analyses of information processing capacities. A) The results of 1d capacity, which corresponds to the memory function for linear memory. B) Total capacity according to the distance and its breakdown in terms of the degree of nonlinearity. All the capacity values classified in the same nonlinear degree are summed up and aligned in a corresponding bar graph.
Even when the computation is transferred, the dynamic process realizing computation does not drop at the active strip but remains to keep the generalized synchronization with the passive one.

The potential of our findings cannot be overestimated. For example, our proposed technique can be applied in emergency situations as a communication tool that transfers a message to a location that humans cannot easily access. This includes communication with people trapped in submerged caves. Another potential scenario can be found in biomedical applications using magnetically guided self-propelled microrobots inducing dynamic self-assembling inside the body. Our conception of self-organized remote reservoirs is, in some sense, similar to the concept of dynamic self-assembling, wherein the system, which is not in equilibrium, shows dynamic pattern formation by permanently receiving and dissipating energy. These robots are expected to act as active reservoirs and interact with passive elements (acting as passive reservoirs) locally inside the body, inducing behavioral change to the robots. For example, it is reported that various dynamic pattern formations can be induced through the interaction of active and passive elements only by changing the morphology of them. If we could externally monitor and read out the dynamics of the self-propelled microrobots interacting with passive materials inside the body, then we can remotely estimate the condition of the internal body using this monitored dynamics as a reservoir which would be helpful for health check. However, implementations of these approaches require further investigations in future.

In our experimental setup, the interaction modality between the two silicone rubber strips can be altered significantly based on the distance between the Active and Passive strips. That is, within a given range of distance (such as 8 cm), the two silicone rubber strips will make direct contact, and if the distance exceeds this range, the strips would only be coupled through the water medium. The difference in the interaction modality strongly affected the information processing capabilities of both silicone rubber strips. This was first confirmed by investigating the performance of Boolean function emulation tasks. Interestingly, many cases were observed in which the reservoir consisting of the Passive silicone rubber strip outperformed that of the Active silicone rubber strip, and further, in some situations, the combination of Active and Passive reservoirs improved performance significantly compared with using either reservoir separately. This result implies that our approach not only transfers computation to a remote location but also can enrich the computational capability of the original reservoir (i.e., Active reservoir) by adding the remote reservoir (i.e., Passive reservoir) and treating the whole system as one reservoir. However, specific interactions sometimes degraded the performance in the reservoir composed of the Active silicone rubber strip, where the behavior of the Passive silicone rubber strip acted as a perturbation to the Active one. Next, by analyzing the expressiveness of the reservoirs in terms of IPC, we revealed that the profile of the capacities differed according to the conditions of the interaction. In particular, we found that contact interaction increases the nonlinear information processing in the reservoirs composed of both Active and Passive silicone rubber strips. Furthermore, linear and long memory was found in the reservoir composed of the Passive silicone rubber strip. These results suggest that by changing the conditions of an interaction, we can induce diverse types of computational capabilities from the system.

Our study has revealed that the boundary of a reservoir is in fact neither static, predefined, nor fixed but can be dynamic, grow, and self-organize using body–environment couplings and generalized synchronization. For example, we might switch the constituent of the reservoir in real time by controlling the strength of the input amplitude in a self-organizing manner. This is especially beneficial in PRC because physical reservoirs are not always easy to redesign, often requiring too much effort and time, compared with software-implemented neural networks. In those situations, using a method to modulate and reconfigure the dynamic properties of the physical system just by externally controlling the input amplitude would allow us to explore the optimal reservoir structure without redesigning the configuration of the physical system according to the given task. Based on a similar motivation, physical reservoirs using a spin-torque oscillator have recently been demonstrated to exhibit input-driven bifurcations, which drastically alter the property of the dynamics from order to chaos and back by only increasing the amplitude of the input stream.

Similarly, echo state networks, which are typical reservoir computing systems, have also been shown to demonstrate a chaos-to-order bifurcation only by increasing the input amplitude, which contributes to improving memory capacity by suppressing chaos. Furthermore, it has been reported that a reservoir system with recurrent infonmax properties (where the real neurons are expected to have this property) tends to self-organize a delay line structure to enhance short-term memory by increasing the intensity of the external input. Further investigations are required to gain perfect command of the control to obtain our required computation in our system. Many free parameters have not yet been investigated in detail, such as the setting related to the input (including the setting of $r$, the amplitude of the input command, etc.), the effect of environmental settings (including the number of remote reservoirs, shape and size of the tank, etc.), or the robustness to external perturbation, all of which are left for future investigations.

4. Experimental Section

**Silicone Rubber Sensor Fabrication**: Silicone rubber was first formed using a mixture of silicone rubber solution (KJR-9060, Shin-Etsu Chemical) and baked at 90°C for more than 1 h. After curing the silicone rubber, CNT was printed using a dispenser (Musashi Engineering) and cured at 90°C for >10 min. Subsequently, galinstan was patterned using the dispenser. In parallel, Ag electrodes were screen printed on a PET film and cured at 90°C for >2 h. This Ag electrode film performed the role of an interconnection between CNT/galinstan and the measurement equipment of a high-speed memory recorder (MR6000, Hioki). After the lamination of PET film over silicone rubber/galinstan film, a silicone rubber was laminated using the silicone rubber solution as a glue and cured at 90°C for >2 h.

**Calculation of Cross Correlations**: Cross correlations indicate a relationship between two data groups. The cross correlation between time series $\{a_i\}$ and $\{b_{i,k}\}$ with the delay $k$ was calculated as follows.

$P_{bi,k} = \frac{\frac{1}{n} \sum_{i=1}^{n} (a_i - \bar{a})(b_{i+k} - \bar{b})}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} (a_i - \bar{a})^2} \sqrt{\frac{1}{n} \sum_{i=1}^{n} (b_{i+k} - \bar{b})^2}}$
where \( \mathbf{b} = \frac{1}{n} \sum_{t=1}^{n} \mathbf{b}_t \) and \( \mathbf{n} \) is the length of the time series, respectively.

The relationships between the time series of Input, Active, and Passive were investigated using cross correlations. The correlation with Input was calculated by delaying Input from 0 to 1500 ms by 3 ms. For the calculation of the correlation between Active and Passive, Active was delayed from \(-1500\) to \(1500\) ms by \(3\) ms.

**Measuring IPC:** To evaluate the information-processing capability of the reservoirs composed of the Active and Passive silicone rubber strips, IPC was calculated.

**Definition of the Boolean Capacity:** IPC quantifies the type and amount of input history held in the system by examining the emulation abilities of orthogonal polynomials. A nonlinear dynamical system receiving an input updated the state through its state equation. Let the input and state be an i.i.d. input \( \zeta_t \) and \( \mathbf{x}_t \in \mathbb{R}^N \), respectively. Under the assumption that the state is a function of only the input history \( \{\zeta_{t-1}, \ldots, \zeta_0\} \) — called an “echo function” — we can predict the orthogonal polynomials of the input history (e.g., \( \zeta_{t-1}, \zeta_{t-2}, \zeta_{t-1}, \zeta_{t-2}, \) and \( 3\zeta_{t-1} - 1 \)) by linear regression using \( \mathbf{x}_t \). Let \( \mathbf{z}_i \) be the \( i \)th multivariate orthogonal polynomial, and we define the prediction capability using the normalized mean square error of the predicted output as

\[
C_i = 1 - \frac{\min \{ (\mathbf{z}_i - \mathbf{w}^T \mathbf{x})^2 \}}{\mathbf{z}_i^2} > (4)
\]

where \( C_i \) is the IPC for the \( i \)th target polynomial, while \( . \) and \( \mathbf{w} \in \mathbb{R}^N \) denote the time average and weight vector of the linear regression, respectively.

To reveal the computational capabilities of binary input-driven systems, we derived the IPC using random binary inputs. If the binary input \( \zeta_t \in \{ -1, 1 \} \) follows the Bernoulli distribution that takes 1 and \(-1\) with probabilities \( p \) and \((1-p)\), respectively, the \( i \)th polynomial is defined by

\[
\mathbf{z}_i = \prod_{j=1}^{n} \left( \zeta_{i-j} - \langle \zeta_i \rangle \right)
\]

where \( n \in \mathbb{N} \) and \( s_i \in \mathbb{N} \) denote the degree and delay step of \( \mathbf{z}_i \) element, respectively. **Table 1** shows an example set of binary polynomials, in which the maximum degree and delay are two and three, respectively. We termed the IPC using binary inputs for the Boolean capacity.

**Property of the IPC:** In this connection, the IPC was equivalent to the squared norm of the coefficient vector of the basis in the expansion of orthonormal polynomials — called the “polynomial chaos expansion.”

We can extract the \( r \)-normalized, linearly independent state \( \hat{x}_t \) from the \( N \)-dimensional state \( \mathbf{x}_t \) through a singular value decomposition. The state was expanded by the polynomial bases of IPC, as follows.

\[
\hat{\mathbf{x}}_t = \sum_{i=1}^{\infty} c_i \mathbf{z}_i
\]

where \( \hat{\mathbf{x}}_t = \frac{x_t - x_t}{\sqrt{C_0}} \) represents the normalized polynomial and \( e_i \) is the coefficient vector for the \( i \)th polynomial. Using the decomposed state, we can rewrite the \( i \)th IPC as

\[
C_i = ||e_i||^2
\]

indicating that the IPC evaluated the norm of the coefficient vector in polynomial chaos expansion as the computational capability. Using a sufficiently large number of polynomials, we composed a complete orthogonal system of \( \{\zeta_{t-1}, \ldots, \zeta_0\} \) and evaluated the computational capabilities without omission. The total capacity \( C_{tot} \), which sums up enough types of capacity, is defined, as follows.

\[
C_{tot} = \sum_{i=1}^{\infty} C_i
\]

Provided that the state is an echo function, the total capacity is equivalent to the rank as follows.

\[
C_{tot} = r
\]

If the state is a function of other variables, except for the input history (e.g., other dynamical noise, measurement noise, and time), \( C_{tot} < r \) because the polynomials no longer comprise a complete orthogonal system for this state.

**Test of Significance:** Because of the finite lengths of time series, the obtained IPCs can include numerical errors, which may increase the value of the measured total capacity. To reduce this accumulated error in the total capacity, we distinguished significantly large capacities, adopting the shuffle surrogate procedure. First, we prepared 200 surrogates that were input series shuffled in the time direction and computed for each capacity using the surrogates. Furthermore, we let the significance level be \( \alpha = 1\% \) and chose the original IPC, which exceeded 1.2 times the value in the top \((\alpha/2)\)% of the 200 capacities calculated using the surrogates. We performed the above operation for each combination of polynomials, obtaining significant IPCs.

**Supporting Information**

Supporting Information is available from the Wiley Online Library or from the author.

**Acknowledgements**

The authors would like to acknowledge Dr. Tomoyuki Kubota for fruitful discussions. This work was supported by JSPS KAKENHI grant numbers JP18H05472 and JP19K14936 and by JST CREST grant no. JPMJCR2014. This work was partially based on the results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization.

**Conflict of Interest**

The authors declare no conflict of interest.

**Data Availability Statement**

The data that support the findings of this study are available from the corresponding author upon reasonable request.
Keywords
embodiments, flexible sensors, physical reservoir computing, soft robotics, synchronization, unconventional computing

Received: August 28, 2021
Revised: October 25, 2021
Published online: December 3, 2021