Graph network-based simulation of multicellular dynamics driven by polymer brush-modified cellulose nanofibers

Chiaki Yoshikawa,[*,†] Duc Anh Nguyen,[‡] Tadashi Nakaji-Hirabayashi,[¶,§] Ichigaku Takigawa,[‖] and Hiroshi Mamitsuka[*,‡]

†*Research Center for Functional Materials, National Institute for Materials Science (NIMS), Tsukuba, 305-0047 Ibaraki, Japan*

‡*Bioinformatics Center, Institute for Chemical Research, Kyoto University, Uji, 611-0011 Kyoto, Japan*

¶*Graduate School of Science and Engineering, University of Toyama, Toyama, 930-8555 Toyama, Japan*

§*Graduate School of Innovative Life Science, University of Toyama, Toyama, 930-0194 Toyama, Japan*

‖*Center for Innovative Research and Education in Data Science (CIREDS), Institute for Liberal Arts and Sciences, Kyoto University, Kyoto, 606-8315 Kyoto, Japan*

E-mail: YOSHIKAWA.Chiaki@nims.go.jp; mami@kuicr.kyoto-u.ac.jp

Abstract

Manipulating the three-dimensional (3D) structures of cells is important for facilitating to repair or regenerate tissues. A self-assembly system of cells with cellulose nanofibers (CNFs) and concentrated polymer brushes (CPBs) has been developed to fabricate various cell 3D structures. To further generate tissues at an implantable

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

level, it is necessary to carry out a large number of experiments using different cell culture conditions and material properties; however this is practically intractable. To address this issue, we present a graph neural network-based simulator (GNS) that can be trained by using assembly process images to predict the assembly status of future time steps. A total of 24 (25 steps) time-series images were recorded (four repeats for each of six different conditions), and each image was transformed into a graph by regarding the cells as nodes and the connecting neighboring cells as edges. Using the obtained data, the performances of the GNS were examined under three scenarios (i.e. changing a pair of the training and testing data) to verify the possibility of using the GNS as a predictor for further time steps. It was confirmed that the GNS could reasonably reproduce the assembly process, even under the toughest scenario, in which the experimental conditions differed between the training and testing data. Practically, this means that the GNS trained by the first 24 hour images could predict the cell types obtained three weeks later. This result could reduce the number of experiments required to find the optimal conditions for generating cells with desired 3D structures. Ultimately, our approach could accelerate progress in regenerative medicine.

Introduction

The main aim of tissue engineering is to repair or improve deficient or damaged tissues/organs.[1] It is essential to generate cells with biologically and physically favourable microenvironments, as well as desirable three-dimensional (3D) structures with designed (anatomical) defect sizes and shapes.[2,3] For this purpose, synthetic or natural biomaterials called scaffolds have been used[4–13] for cell attachment, due to the fact that they provide an appropriate space for cell migration and promote the diffusion of nutrients/oxygen. In addition, scaffolds improve waste release, are biocompatible and act as robust structures until the cells are able to form tissues/organs. Over the last three decades, various scaffolds such as gels, sponges, porous materials, and fibres have been extensively developed.[11–15] Although thin sheet-type tissues,

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

such as the retina and skin, have been successfully regenerated,[16–18] the regeneration of larger and more complicated tissues/organs has not yet to be realized.

During development of the human body, stem cells autonomously assemble with the extracellular matrix (ECM) to hierarchically form the sophisticated 3D structures of tissues and organs.[19] Inspired by this process, a cell self-assembly system using cellulose nanofibers (CNFs) has been developed to fabricate various 3D structures.[20–23] These CNFs served as an artificial ECM, due to the fact that their dimensions (3 to 20 nm in diameter and several micrometers in length) are analogous to those of collagen, i.e. the main component of ECM.[24–27] To introduce a driving force for self-assembly of cells and CNFs, the CNF surfaces were modified with concentrated polymer brushes (CPBs). The CPBs obtained via surface-initiated living radical polymerization showed the highest graft density reported to date (dimensionless graft density >0.1).[28] In good solvents, the CPB polymer chains were highly extended, nearly to the full length, due to the high osmotic pressure.[28] As a result, the CPBs exhibited highly repulsive interactions, leading to a good dispersion of colloids and the inhibition of nonspecific protein adsorption and cell adhesion.[28–34] In more recent studies, concentrated poly p-styrenesulfonic acid sodium salt (PSSNa) brushes were grafted onto CNFs, due to the relatively bioinert character of the anionic PSSNa and its ability to weakly associate with cells via electrostatic forces.[20–23] The resulting CNF-PSSNa dispersed well in the cell culture medium and reversibly assembled with human mesenchymal stem cells (hMSCs).[23] Notably, hMSCs self-assembled with CNF-CPBs in chondrogenic differentiated media to form a single unique structure, namely a giant sheet with dimensions on the millimeter scale. In addition, using the CNF-CPB structure enhanced chondrogenesis compared with the cell pellets (i.e. the gold standard for stem cell culture). These findings strongly indicate that even in vitro, tissues or organs can be regenerated by self-assembly of cells and artificial ECM, such as CNF-CPBs.

Although the above self-assembly system can be considered useful for tissue engineering applications, the regeneration of tissue and organs on a practical/implantable level requires

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

hundreds and thousands of trial-and-error experiments to be conducted. Moreover, it is necessary to select useful information from huge amounts of data and feed it back into the next experiment; this is practically impossible because of time, manpower and consumer limitations.

To overcome such difficulties in experimental research, machine learning (ML) has attracted growing attention as a powerful tool that can quantitatively estimate the relationships between experimental conditions and the obtained results, whilst also autonomously discovering hypotheses hidden in the data.[35] There already exist a lot of traditional ML algorithms, such as linear regression, and the appropriate algorithms should be selected, depending on the data type and the target application. Recently, deep learning (DL) has become a focal point due to its higher classification performance than conventional ML techniques when there is a large amount of data, particularly in terms of processing images or videos from biological experiments.[36]

In the last decade, convolutional neural networks (CNNs), which are considered the leading DL model,[37,38] have been successfully applied to various cell-related applications, such as the processing of image data to track cell movement,[39] identifying cell features,[40,41] and predicting the differentiation of organoids.[42,43] It was considered that DL may be beneficial for the above self-assembly system due to the collection of live cell imaging (video), tissue section images and confocal micrograms to evaluate cell functions and cell differentiation. However, the above system is more complicated, since it contains not only cells but also CNF-CPBs. Therefore, the proposed system requires an elaborate algorithm to allow its implementation.

The procedural feature of DL is information propagation through a layered network, where one node in a layer can be connected to (usually) all nodes in the next layer (Fig. 1 (a)). This node connection can be regularized using prior knowledge, for which a pre-defined graph can be used. In this graph, an edge between two nodes indicates a high similarity between the two nodes. In other words, instead of connecting to all nodes in the next

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

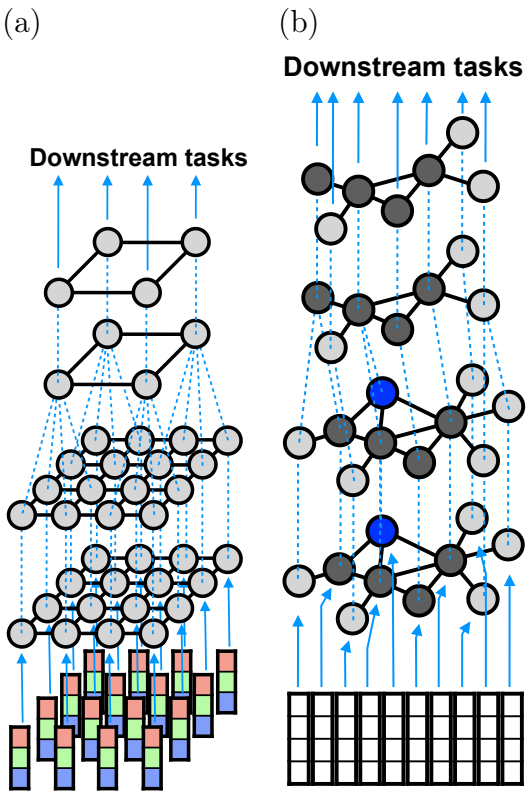

Figure 1: Two DL models: (a) Convolutional Neural networks (CNN) and (b) Graph convolutional neural networks (GNN).

layer, the node connections can be limited using the pre-defined graph (Fig. 1 (b)). This is otherwise known as a graph neural network (GNN), where the node connections are more flexible (due to the regularization by a given graph) than those where all connections between two layers are established in the CNN.[44] As a result, GNN can regularize the information propagation of DL, and is thought to be a generalization of the CNN (Fig. 1).

ML can be used for predicting the labels of unknown examples. In time-series data, it can predict the label at the next time stamp. Thus, a dynamic system can be achieved, such as a physical particle system, using a graph where a node represents a particle and an edge represents the interaction between two particles. In a GNN, the information is propagated through the edge connecting the two nodes (particles).[45] Once a GNN is trained based on the past behavior of the system, the trained GNN can be used to show the future behavior of the corresponding system. This prediction by a trained GNN is a type of simulation, which

https://doi.org/10.26434/chemrxiv-2023-9ll6s ORCID: https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. License: CC BY-NC-ND 4.0

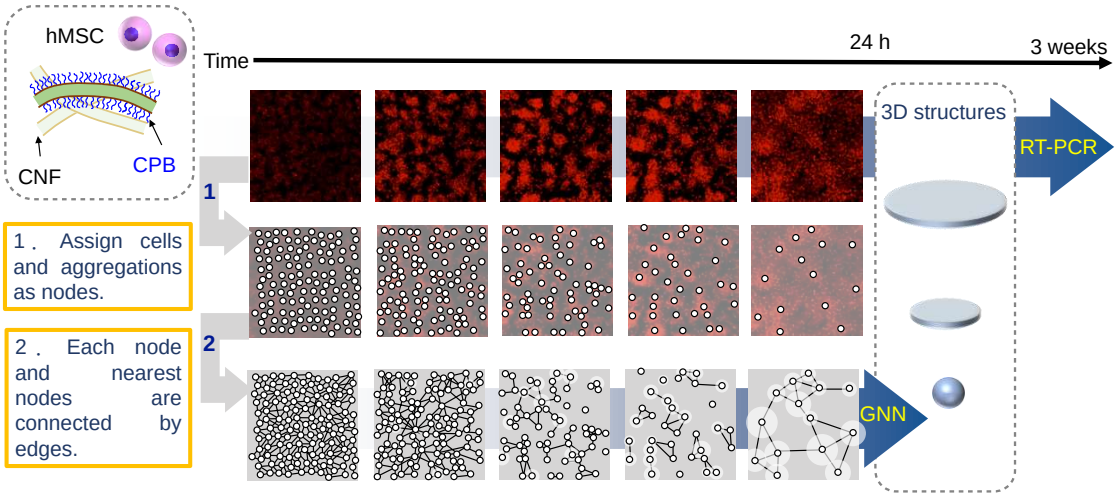

Figure 2: Schematic representation of data generation for the GNN-based simulator (GNS).

allows the generation of the data for further training. The information propagation of DL can be formulated with prior knowledge, such as Newton's laws of motion, which facilitates the implementation of DL or GNN simulation for time-series prediction. It has been previously demonstrated that a GNN is useful for simulating systems in physics.[45] More specifically, a GNN can reproduce the results of a physical simulation model when it is trained using data generated from the simulation model, and the GNN itself can be a graph network-based simulator (GNS). However, this GNS has not yet been applied to real biological systems which tend to be more complex than physical systems.

With the above consideration in mind, the objective of this study is to investigate whether the GNS could simulate the dynamics of biological cells over time (see Fig. 2). It was considered that using recorded time-series cell aggregation images, it should be possible to manually identify cells in each image as nodes in a graph (generated by connecting nearest nodes by their edges). This process should generate the times-series, dynamic graphs, which could subsequently be used to train data of a GNS.[45] For the purpose of this study, three different scenarios (experimental settings) were considered for combination of the training and test data. Scenario I verified the performance of predicting later time frames (as test data) by using earlier time frames in the same file (as training data). The favorable predictive

https://doi.org/10.26434/chemrxiv-2023-9ll6s ORCID: https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. License: CC BY-NC-ND 4.0

performance result of the GNS for Scenario I uncovered the high predictability of a GNS for later time frames by using earlier time frames within the same file. Scenario II examined the problem of predicting time frames (as test data) using the time frames of other files of the same environment (as training data). In this case, if the training is started from later time frames of a test file, the prediction result by the trained GNS showed a similar high performance to Scenario I. This indicates that later time frames can be predicted using other files with the same property. Scenario III used the time frames of other files of various (different) environments (as training data). The performance result of the trained GNS in Scenario III was pretty similar to Scenario II, implying that later time frames can be predicted using those of the other files obtained under different environments from the test file.

These results indicate that the trained GNS with only 24-hour data could predict the future aggregation of CNF-CPBs, i.e. the cell types obtained later by cell differentiation of more than three weeks (Fig. 2). This result shows that applying a GNS to simulating time-series multicellular images is very promising and innovative to drastically reduce the number of repeating heavily time-consuming experiments. We emphasize that this work is totally new in the relevant fields, in terms of applying machine learning to reducing the cell differentiation experiment costs.

This time, our data was generated with only three environments and four repetitions. It can be expected that by feeding a larger number of samples (repetitions) obtained under more various conditions/environments (with more detailed time intervals) into a GNS, the trained GNS would be more robust and have higher predictability. The trained GNS will be able to be used as a simulator of our self-assembly cells to find the most optimal conditions/environments to generate a particular type of self-assembly cells, such as sheet-type tissues. In other words, after training, the trained GNS can be run many times under various conditions (environments) and it is possible to select the condition, under which the trained GNS can give the most similar result to the designated cell assembly type: the selected

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

condition would be the most suitable one to generate the target type of cell assembly (or cells with a particular 3D structure).

Experimental Settings

Materials

Cu(I)Br (99.99%, FUJIFILM Wako Pure Chemical Corp., Osaka, Japan), Cu(II)Br2 (99.99%, Wako), 2,2'-bipyridine (bpy, 99.9%, Nacalai Tesque, Inc., Kyoto, Japan), poly(ethylene glycol) methyl ether 2-bromoisobutyrate (PEGBr, Merk Japan, Tokyo, Japan), and p-styrenesulfonic acid sodium salt (SSNa) (99.9%, Merk Japan) were used for the purpose of this study. The CNF (WFo-100, approximately 2 wt% in an aqueous slurry) was purchased from Sugino Machine, Ltd., Toyama, Japan. A PSSNa standard (Polysciences, Inc., PA, USA) was used for the gel permeation chromatography (GPC) system.

Synthesis of CNF-Br

The CNFs bearing an ATRP initiator (CNF-Br) were prepared by esterification of the purchased CNFs with 2-bromoisobutyryl bromide (kindly donated by Dr. K. Sakakibara, National Institute for Advanced Industrial Science and Technology). Further details of the preparation followed.[20–23]

SI-ATRP of SSNa

SI-ATRP reaction between SSNa and the prepared CNF-Br was performed as described previously.[20–23] The number-average molecular weights (Mn) and polydispersities (Mw/Mn) of the free polymers were determined using a PSSNa-calibrated GPC system. The conversion was determined using ^{1}H nuclear magnetic resonance (NMR) spectroscopy. The amount of PSSNa grafted onto the CNF was estimated by elemental analysis, as detailed previously.[20–23]

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Following the polymerization reaction, the obtained CNF-PSSNa was washed with Milli-Q water, and the concentration was adjusted to approximately 3 wt%. Aqueous solutions of the CNF-PSSNa were stored at 4°C until required for use, and water was exchanged with the cell culture medium before cell seeding. The characteristics of the prepared CNF-PSSNa are outlined in Table S1.

Cell culture

Normal human bone marrow-derived mesenchymal stem cells (hMSCs) (multiple donors) were purchased from LONZA (Switzerland). These hMSCs were maintained in basal growth medium (MSCGM BulletKitTM) (LONZA PT-3001) at 37°C in a humidified air environment containing 5% CO_2. Upon reaching subconfluence, the cells were harvested from the flasks using trypsin. The cells were used at the fifth passage.

Cell culture with CNF-PSSNa

A suspension of the hMSCs (0.5 mL 1 $\times$ 10^6 cells/mL) wax mixed with the prepared CNF-CPBs (0.5 mL, 0.2, 0.1, or 0.01 wt%). Subsequently, the cell suspension (1mL, 5 $\times$ 10^5 cells/mL) containing CNF-PSSNa (0.1, 0.05, and 0.005 wt%) was placed in a low-attachment 24-well plate (PrimeSurfaceTM Plate 24F, Sumitomo Bakelite Co., Ltd., Japan) and cultured for the desired duration. For chondrogenic induction, a chondrogenic induction medium (hMSC BulletKitTM, LONZA PT-3003) containing recombinant human transforming growth factor beta-3 (rhTGF-β 3) was used. The basal and chondrogenic differentiation media were changed every 2–3 days, following the instructions provided by LONZA. As a control, only hMSCs were cultured in low-attachment 24-well plates (PrimeSurfaceTM Plate 24F). An aliquot (1mL) of the cell suspension (5 $\times$ 10^5 cells/mL) was added to each well.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Live cell imaging

The self-assembled hMSC structures were monitored using a fluorescence microscope equipped with a cell incubation chamber (5% CO_2 at 37°C, All-in-one Fluorescence Microscope BZ-X 800, KEYENCE Co. Ltd., Osaka, Japan). Prior to observation, the hMSCs were stained with a PKH26 red fluorescent cell linker kit (MERCK) according to the manufacturer's protocol. The stained cells were cultured with or without CNF-PSSNa and maintained in the microscope chamber for 24 hours. Images were captured every 30 minuntes. Table S2 summarizes the experimental conditions and abbreviations employed when referring to the GNS dataset.

Cell tracking

The cells were tracked manually using ImageJ software. More specifically, in each image, the location of each cell was identified, and each identified cell was traced using time-series images.

Graph generation

Each cell in an image was regarded as a node, and the nearest nodes were connected by edges, resulting in a single graph for one image. For the time-series images, each node was traced by cell tracking using the time-series graphs, as described above.

Entire data

Fig. 3 illustrates the entire data for the various time frames of the biological cells. Two experiments (EXP) were carried out, namely HMSC (H) and Chondro (C). Each experiment contained three datasets (ENV), each corresponding to one of the three environments, namely the Control (C), Low (L), and High (H) environments. A single dataset contained four files (repetitions), and each file contained 25 time frames. Each data file was named using the

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

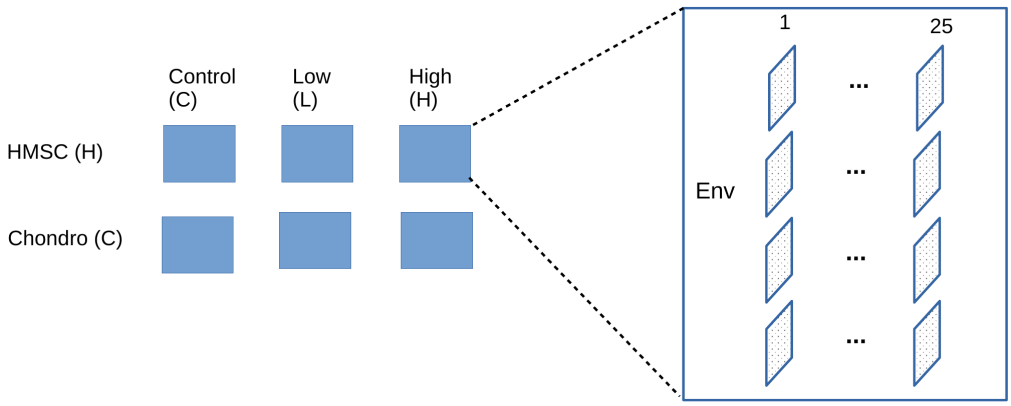

Figure 3: Schematic representation of the entire datasets.

format {EXP}{ENV}_{FileID}, where {EXP} can be either H for HMSC or C for Chondro, {ENV} can be either C for Control, L for Low or H for High, and {FileID} can be either 0, 1, 2 or 3. For example, CH_0 stands for the first data file (0) of the Chondro experiment under the High environment.

Graph network-based simulator (GNS)

A graph network-based simulator (GNS) was used (it was previously proposed to simulate particle systems in physics[45]). The basic hypothesis of a GNS or, more generally, a graph neural network is that, given a set of time-series data points (corresponding to nodes in a graph), the next state of a point can be determined based on the information associated with the nearest neighbor points of the current state. This information transfer process is known as 'message passing'. In a GNS for simulating particle systems in physics, message passing is represented by Newtons' laws of motion, and the parameters of these laws are trained using given data. In the prediction (simulation), the information of the current position is affected by the neighboring points of the current position (through interaction forces). Fig. 4 illustrates this framework.[45] Given an input with the state of the data point (particle) positions at the current step (t_k), the graph network outputs the state of point positions at the next step (t_{k+1}). Here, the next state of a point is affected by its neighboring points of the current position (through interaction forces), which can be divided into three stages:

https://doi.org/10.26434/chemrxiv-2023-9ll6s ORCID: https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. License: CC BY-NC-ND 4.0

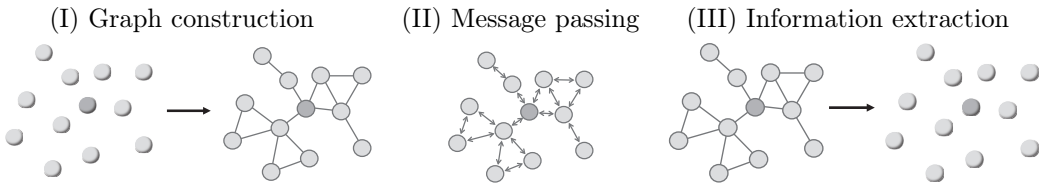

Figure 4: Illustration of the graph networks for the simulations.

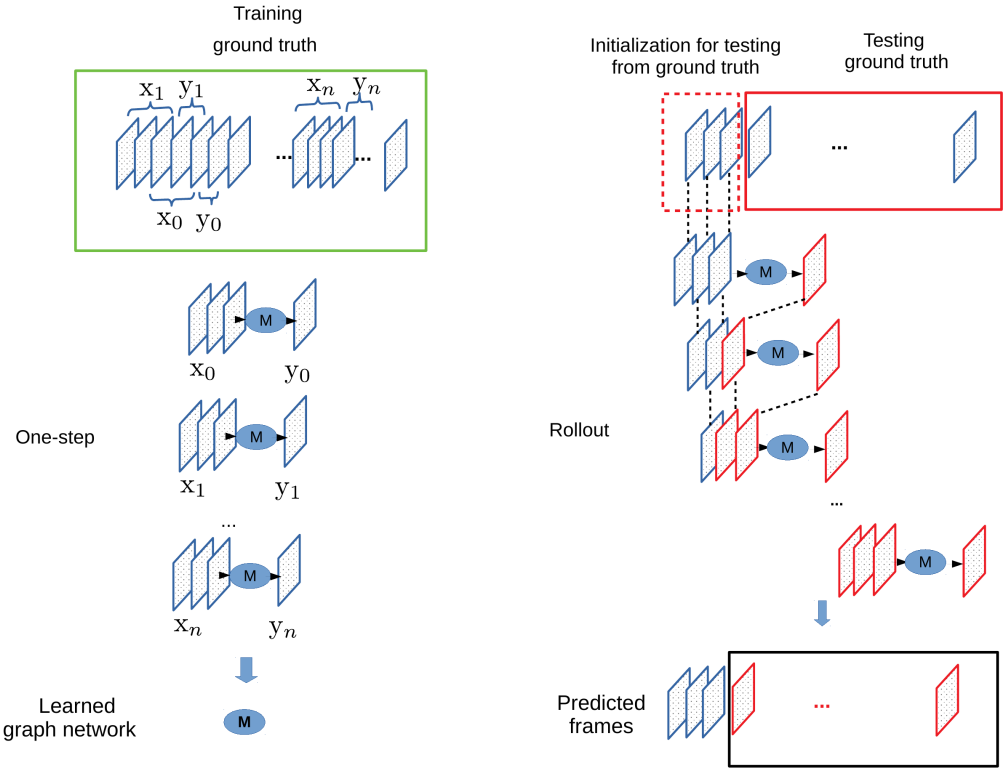

Figure 5: Illustration of the training and testing procedures for the simulation with graph networks.

(I) constructing a graph, (II) passing messages, and (III) extracting dynamic information. In stage (I), a graph is created with nodes for the points and edges for pairs of neighboring nodes. In stage (II), the information from the neighboring nodes is propagated to update the next state of each node. Finally, in stage (III), the corresponding positions of data points in the next state are extracted.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Training and testing procedure

Fig. 5 illustrates the procedure of training model M (left) and of measuring the performance of the simulation obtained by running the trained M (right) over a given sequence of time frames for training and testing. In training (left of Fig. 5), model (M) is trained by *one-step training*, where one example is a pair (x_i, y_i) of arbitrary N sequential time frames x_i (from the training data) for the input and the next frame y_i for the ground truth output (N=3 in Fig. 5 and N=6 in our experiments). Model M is updated to minimize the loss between the predicted output and the ground truth. This one-step training protocol is repeated until the loss is converged. In testing (right of Fig. 5), each time-frame output is predicted using the previous continuous N frames and the trained model M. This prediction is repeated from left to right of the given test file. This process is called *rollout testing*, which consists of two key points: 1) the N frames can start with any frame of the test file, meaning that the first T frames can be skipped. 2) the first N frames are the ground truth frames in the test data file, and thus when the next N frames are used, the last of these N frames is not the ground truth but the predicted one, meaning that except for the first N frames, predicted frames are always used instead of the ground truth, when available. This point will be more well illustrated in Fig. 8.

Three experimental procedures (scenarios)

Fig. 6 shows the three possible scenarios used to validate the performance of the GNS. The difference between these scenarios is the procedure used to divide the data into training and testing.

- Scenario I: the first part of each dataset is used for training, and the rest is used for testing. The idea behind this scenario is to check whether the GNS can simulate later time frames using earlier time frames in the given data. In the experiments, the first 20 frames are used for training, and the remaining five frames are used for testing.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

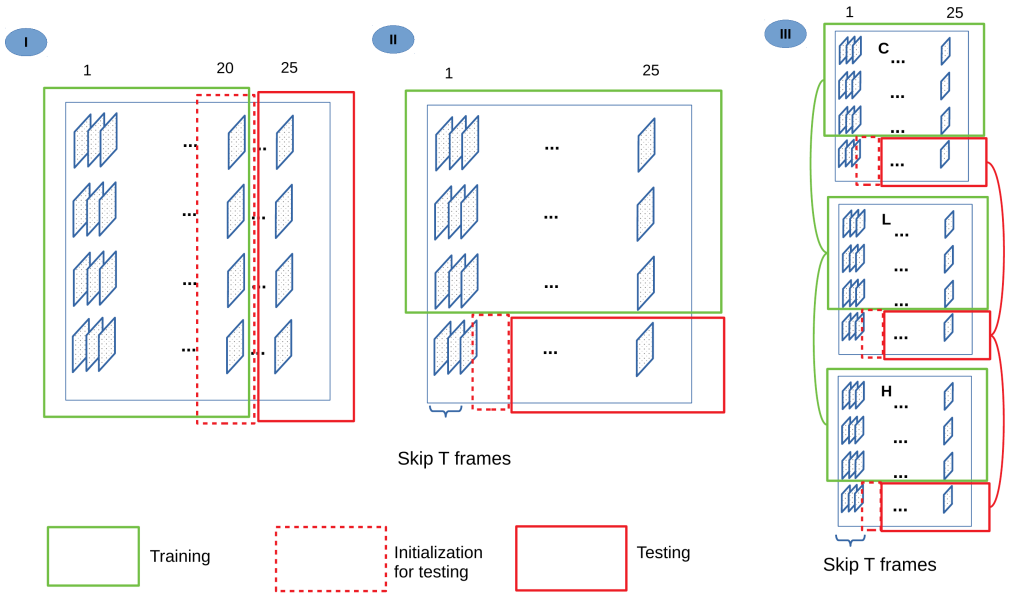

Figure 6: Three scenarios used for the experiments.

- Scenario II: For each dataset, all files are used for training, except for one file, which is used for testing. One-step training is used for the training files and rollout testing is used for the test file. The idea behind this scenario is to check whether the GNS can simulate time frames of a file that are different from those used for model training. In the experiments, three files are used for training and the remaining file is for testing.

- Scenario III: This scenario is an extension of Scenario II with multiple datasets (only one dataset is used for Scenario II). The idea behind this scenario is the same as Scenario II. In the experiments, three datasets, namely Low, High and Control (i.e. 3×3 files together) are used for training. After model training using these datasets, each frame of the remaining file of each dataset is predicted.

Performance evaluation measure

To evaluate the prediction performance in terms that a predicted frame is close to the corresponding target ground truth frame, an appropriate measurement protocol is required. Since this protocol should measure how cells (nodes) are grouped (or aggregated) in the time-series process, it is not necessary to check whether the exact location of the ground

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

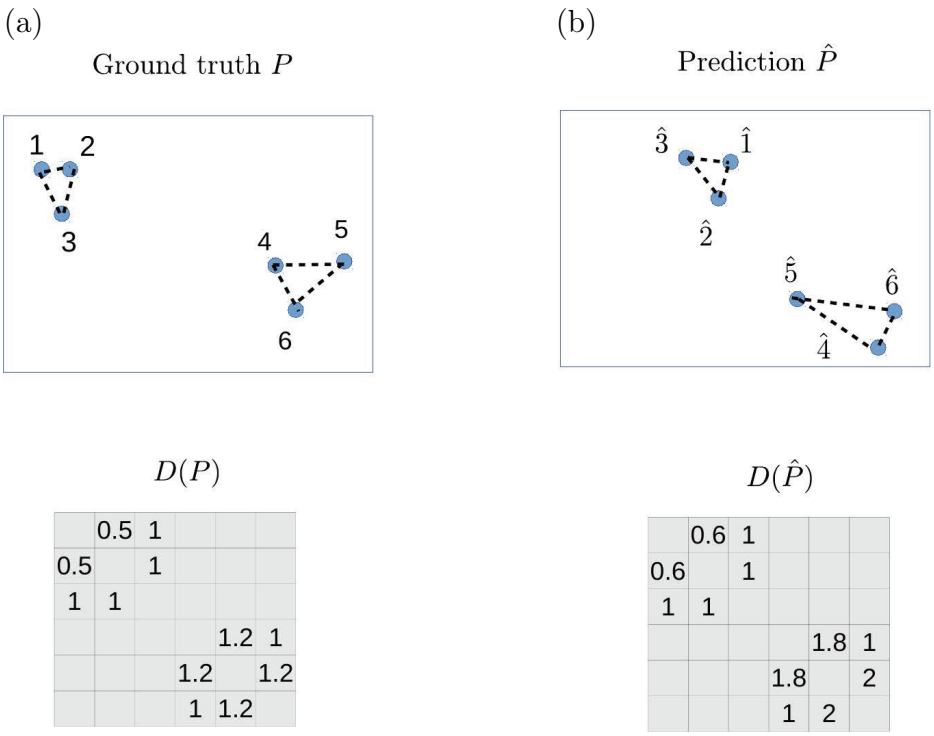

Figure 7: Illustration of comparing the distance matrices of the neighbor nodes.

truth is kept in the prediction. Instead, it is necessary to check whether the properties of the grouping cells in the ground truth are captured by the simulation. Hence, a traditional measurement approach, such as the mean squared error (which is based on the idea that the predicted position should be the same as the ground truth position) is not suitable.

The upper panel of Fig. 7 shows a simple, illustrated sample to explain the above idea, where two images, i.e. the ground truth frame P and the predicted frame $\hat{P}$, are shown. The predicted frame $\hat{P}$ has two groups of data points: $(\hat{1}, \hat{2}, \hat{3})$ and $(\hat{4}, \hat{5}, \hat{6})$, which are consistent with the two groups of data points in the ground truth: $(1, 2, 3)$ and $(4, 5, 6)$. This example should be a good prediction, since the two groups are maintained in the prediction, although the positions of the six data points are not necessarily consistent. Thus, the evaluation metric should provide a good score to this example. The evaluation can therefore be conducted using the point distance matrices $D(P)$ and $D(\hat{P})$ (see the lower panel of Fig. 7 for an example) for the ground truth and the prediction, respectively. In other words, the consistency in the grouping behavior between the ground truth and prediction should be measured using the

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

correlation between these two matrices. Formally, the correlation measure S between the ground truth and prediction can be defined as follows. Let $P = \{\mathbf{p}_i \in \mathbb{R}^2, i = 1...N\}$ and $\hat{P} = \{\hat{\mathbf{p}}_i \in \mathbb{R}^2, i = 1...N\}$ be the two sets of positions of N points in the ground truth and predicted frames, respectively. The correlation measure is

$$\mathrm{S}_k(P, \hat{P}) = Corr(D_k(P), D_k(\hat{P})), \quad \text{where} \tag{1}$$

$$D_k(P)_{i,j} = \begin{cases} ||\mathbf{p}_i - \mathbf{p}_j||_2 & \text{if } i \in \text{ top } k \text{ nearest neighbors of } j, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

In the experiments, Pearson correlation[46] is used for the correlation function $Corr$ and the number of the nearest neighbors, k, is set at 4. The value of this measure is in between -1 and 1, where prediction is better as this measure is closer to 1.

Gene expression analysis

After the live cell imaging, the cells both with or without CNF-PSSNa were continuously cultured for 21 days, during which the cell culture medium was changed every 2-3 days. At 21 days, the cell RNA was extracted and purified using the Qiagen RNeasy Mini kit (QIAGEN) according to the manufacturer's protocol. The extracted RNA (15 ng) was subsequently reacted with the PrimeScript RT reagent kit (TAKARA) at 37°C for 15 minutes and 85°C for 5 seconds to synthesize the desired cDNA via reverse transcription. The quantitative reverse transcription polymerase chain reaction (RT-qPCR) was subsequently conducted on the target genes using a LightCycler® 480 SYBR Green I Master (Roche, Germany). The housekeeping gene glyceraldehyde-3-phosphate dehydrogenase (GAPDH) was used as internal standard. The primers used for the RT-qPCR experiments are listed in Table S3. Expression of the target genes was normalized to that of the housekeeping gene (GAPDH).

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Statistical analysis

Statistical analysis was performed using the Prism 8.2 software package (GraphPad, San Diego, CA, USA). For the three week samples, gene expression is presented as the mean ± standard deviation (SD) of three samples. One-way ANOVA and Tukey pairwise comparisons were used to analyze the statistical differences between the data. Significance was determined at levels of *p <0.05, **p <0.01, ***p <0.001, and ****p <0.0001.

Other measurements

GPC analysis of the PSSNa was performed on a Shodex GPC-101 instrument (Tokyo, Japan) equipped with two Shodex gel columns. A flow rate of 0.8 mL/min was employed using a water/acetonitrile (6:4) mixture with 10 mM LiCl as the eluent (40°C). The column system was calibrated using the PSSNa standard. ^{1}H NMR spectroscopy measurement was performed using an ESC-400 spectrometer (JEOL, Japan). Elemental analyses (EA) were performed at the Microanalytical Laboratory of the Institute for Chemical Research, Kyoto University.

Results and discussion

Wet-lab experiments

Self-assembly of hMSC with CNF-CPBs

PSSNa was initially grafted onto the CNFs by ATRP. The characteristics of the resulting CNF-PSSNa are listed in Table S1. Due to the criteria of CPB (σ^* >0.1),[28] the PSSNa brush was categorized as a CPB. The CNF-CPBs were mixed with hMSCs at different concentrations (0, 0.0005, and 0.01 wt%). The cell mixtures were cultured in basal medium or chondrogenic induction media. It was previously reported that in a basal medium, the combination of hMSCs with CNF-CPBs generated small flocs that increased in size upon

https://doi.org/10.26434/chemrxiv-2023-9ll6s ORCID: https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. License: CC BY-NC-ND 4.0

increasing the incubation time.[23] In contrast, in a chondrogenic induction medium, a single self-assembled structure (a sheet or a ball) was generated, where their shape and size were kept relatively constant after 24 hours. For the GNS, each cell was treated as a particle, and the cell dynamics were observed for 24 hours, when the cells were slightly elongated. To simplify cell tracking, the concentrations of the hMSCs and the CNF-CPBs were reduced to one-tenth of those used in.[23] Prior to cell culture, the hMSCs were stained with fluorescent dyes for live cell imaging. Fluorescent images were acquired every 30 minutes over a 24 hour period, although the majority of cells had disappeared from the frames after 12 hours. With these results in mind, the GNS was operated using cell dynamics frames recorded over the initial 12 hour period (25 frames per sample).

After observing the cell dynamics for 24 hours, the hMSCs cultured both with and without the CNF-CPBs in a basal- or chondrogenic induction medium were maintained in a typical CO2 incubator for three weeks. Photographic images and phase-contrast micrograms of the self-assemblies (flocs) are shown in Figs. S1 and S2, respectively. It can be seen that the hMSCs cultured with 0 and 0.0005 wt% CNF-CPB formed spheroids (balls) independent of the medium used. This result is consistent with that of.[23] In contrast, different from the previous result (a single sheet), torn sheet like-structures were observed in the presence (cell culture) of 0.01 wt% (CH) CNF-CPB, i.e. one-tenth in the cell culture concentration, which prevented the association of the scattered small sheets to form a single sheet.

hMSC chondrogenic gene expression

After three weeks of incubation, chondrogenesis of the cells was evaluated by RT-qPCR of type I collagen (*COL1*), type II collagen (*COL2*), aggrecan (*Aggrecan*) (Fig. S3). All genes were normalized to GAPDH, a ubiquitous housekeeping gene. *COL2* and *Aggrecan* are important for cartilage formation, while *COL1* is typically observed in undifferentiated hMSCs. To confirm that cell staining with fluorescent dyes does not affect the chondrogenesis, non-stained hMSCs were also cultured with CNF-CPBs. As shown in Fig. S4, the trend

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

of upregulation for *COL2* and *Aggrecan* was similar to that presented in Fig. S3, indicating that the fluorescent dyes did not affect the cell functions.

In the system cultured with 0.01 wt% CNF-CPBs (CH samples), remarkable upregulation was observed for the expression of *COL2* and *Aggrecan* (compared with 0 and 0.0005 w% CNF-CPBs (CC and CL, respectively)). Since the *COL2/COL1* expression ratio is known to be relatively high when the hMSCs differentiate into hyaline cartilage, this ratio was evaluated, and it was found that the CH samples showed dramatically higher *COL2/COL1* ratios than the CL and CC samples (0.0005 and 0 wt% CNF-CPBs). This result is consistent with,[23] indicating the suitability of using live cell imaging data for the GNS.

Computational simulations

Below, for each scenario, the following three items will be described: 1) a detailed experimental procedure, 2) prediction performances (Pearson correlations) of the last frames of all test cases, and 3) prediction performances over time-frames of the good and bad cases with links to movies[1]. Additionally, for scenario I, one more item on a data property, which can explain simulation performance, particularly poor-performance cases, is described.

Scenario I

1) Detailed experimental procedure. Fig. 8 outlines the procedure for Scenario I. For each dataset, the first 20 frames of the four files were used as training data, and the remaining five frames as test data. After obtaining the trained model M by one-step training over the training data, rollout testing was performed to predict the remaining frames, given the initial frames as the last frames of the training data (three frames in Fig. 8, and six frames in our experiments).

2) Evaluation of the last frames. Fig. 9 shows the prediction performances of all 24 test cases for the last (25th) frames. There were only two cases (HH_1 and HL_3), in which

[1]The full code and visualization are available at `https://www.dropbox.com/sh/o19f9k8ztv6adtp/AADTaDKqHbXErDOgXDppjxMBa?dl=0`

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

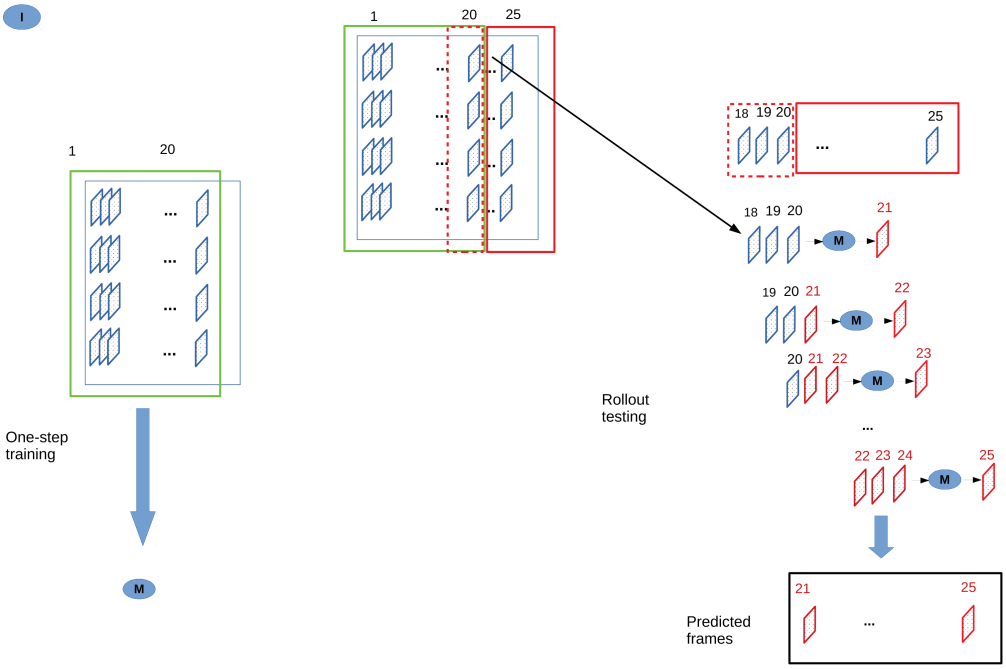

Figure 8: Training and testing in Scenario I.

the performances were below 0.85, and approximately half of the cases had performances of above 0.9. These results clearly indicate that the simulation performance of this scenario looks entirely favorable.

3) Evaluation on good/bad cases over various time steps. Fig. 10 shows the performance evaluation results for the predicted frames over time for the best (CH_0 and CH_3: solid lines) and worst (HH_1 and HL_3: dotted lines) cases. The performances of the best cases were found to be stable and high (above 0.97) over all time frames, whereas for the bad cases, the performances decreased significantly over time frames.

Animations are available for both the good and bad cases: CH_0 at Movie I: CH_0 and HH_1 at Movie I: HH_1.

4) A data property can explain the performance. Based on the above movies, it was hypothesized that if the test frames exhibit large changes in the cell positions, the prediction performance is likely to be worse. Fig. 11 shows an illustration of this hypothesis. Let $\vec{v}^{(t)}$ be the velocity (change speed of the position) at step (t). Assuming that the size of $\vec{v}^{(t)}$ and that of $\vec{v}^{(t-1)}$ are similar, if the size of $\vec{v}^{(t+1)}$ is larger than them, this means a larger jump.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

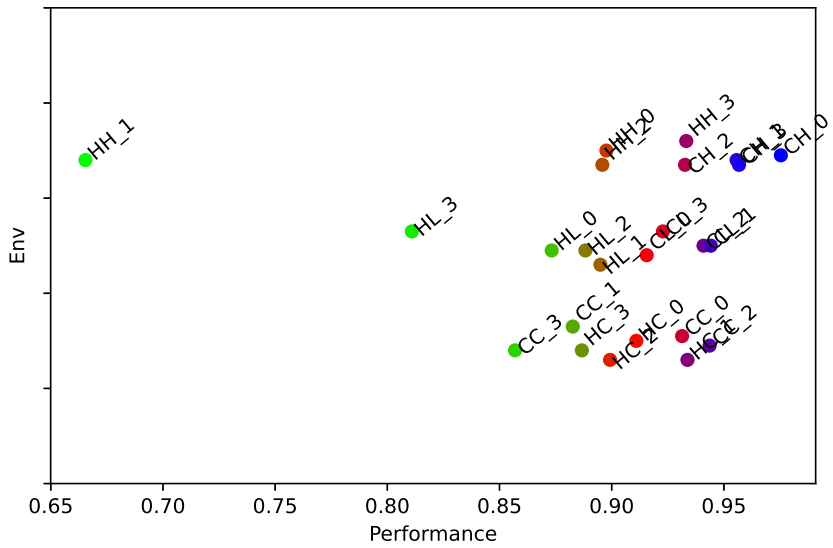

Figure 9: Prediction performances of Scenario I on the the last frame of all test cases.

This change in velocity means that the acceleration $\vec{a}^{(t+1)}$ is larger than the acceleration $\vec{a}^{(t)}$. Our hypothesis is that if the test frames have larger jumps than the given training frames, the prediction will be less accurate.

We examined this hypothesis using real results below. First, the velocity and acceleration can be computed as follows:

$$\mathbf{v}^{(t)} = \{\mathbf{p}_i^{(t)} - \mathbf{p}_i^{(t-1)} |\ i = 1, ..., N\}$$

$$\mathbf{a}^{(t)} = \mathbf{v}^{(t)} - \mathbf{v}^{(t-1)}$$

Subsequently, the mean and standard deviation were measured for the acceleration of the training frames [1:20] and the test frames [21:25]. Fig. 12 shows the mean and standard deviation of the training and test frames of all data files, where the solid and dotted lines represent the training and test data, respectively. It can be seen that the data files on the more right-hand side achieve higher prediction performances. In addition, data files with higher prediction performances (above 0.95) are likely to show smaller means of acceleration. On the other hand, the lower performance data files (less than 0.9) are likely to show larger

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

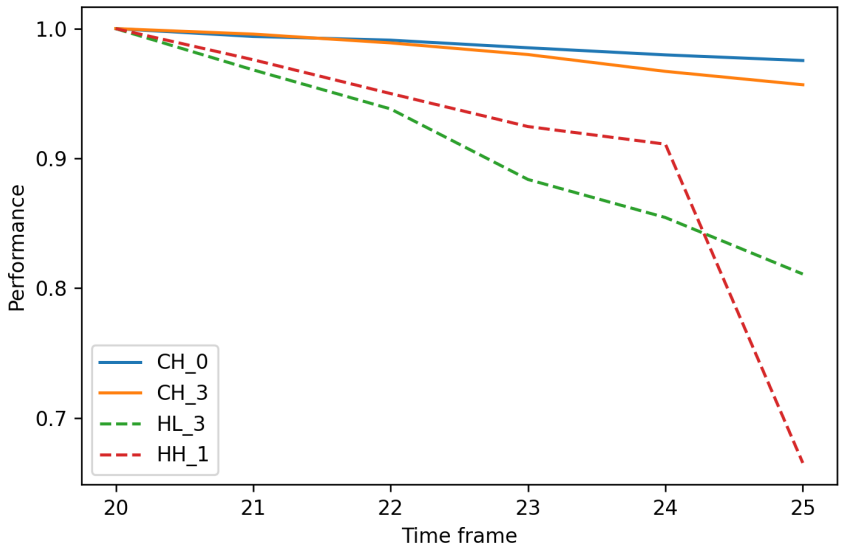

Figure 10: Prediction performances of the good/bad cases in Scenario I over different time frames.

means of acceleration, particularly in test data (dotted lines). These results indicate that higher predictive performance can be obtained when 1) the data files have stable changes (without large acceleration) and 2) the content of training frames covers that of the test frames. Finally, it therefore appears that smaller time steps and a larger number of training frames may be required, because of the above 1) and 2), respectively.

Scenario II

1) Detailed experimental procedure. Fig. 13 shows a detailed procedure of Scenario II. For each dataset, three files with all frames [1:25] were used to perform one step training. Rollout testing was then conducted, starting with an arbitrary time frame in the remaining file. In general, it is expected that the cell behavior at the beginning of the 25 frames would not have any clear pattern of cell grouping. Thus the starting point can be a time frame at a certain number (T) of frames away from the beginning (if T is 9, the prediction starts with the 10th frame). Thus, T is a parameter, which might change the performance, and the performance variation upon changing T was evaluated.

2) Evaluation on the last frames. Fig. 14 shows the prediction performances on the last

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

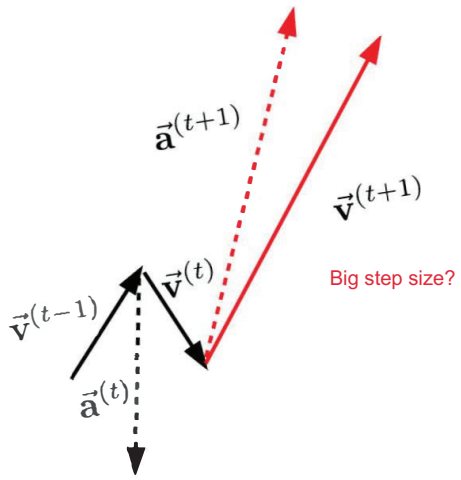

Figure 11: Illustration of a large jump with a large step size for a large acceleration $\vec{a}^{(t+1)}$.

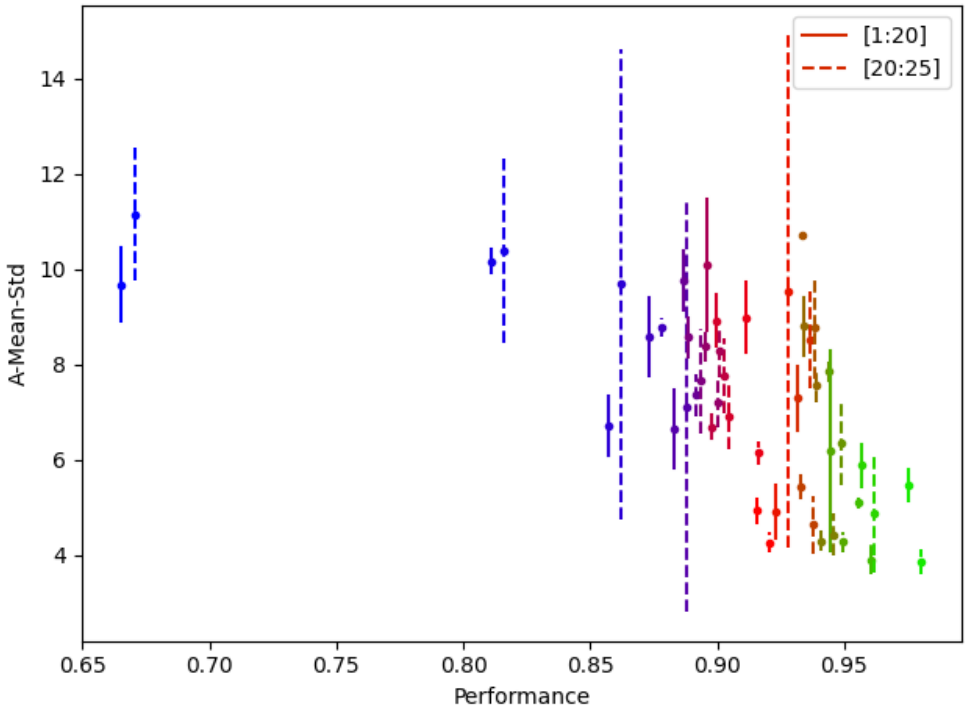

Figure 12: Accelerations and prediction performances.

frames of all test cases with different values of T (0, 4, 9 and 14). In this figure, each file is indicated by the same color (for different values of T). In general, the performance was found to increase as T increased, mainly due to the fact that larger values of T led to a decrease in the number of predictions until reaching the last frame. In particular, for T of 14 (the first

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

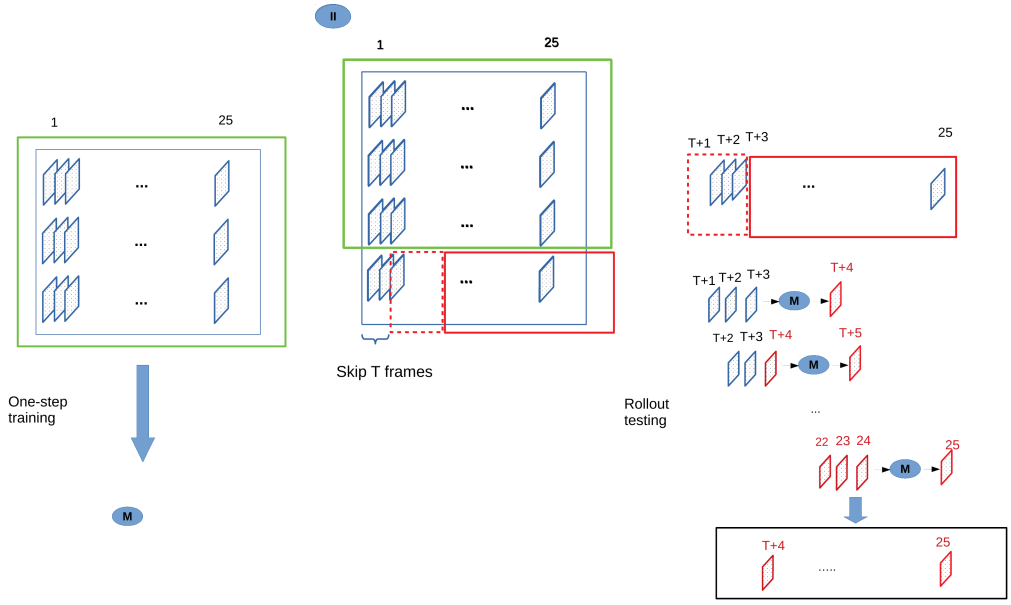

Figure 13: Training and testing in Scenario II.

frames used for prediction were the 15th to the 20th and the predicted frames were the 21st to the 25th), prediction performances were mainly in the range of 0.85 to 0.95, which are comparable to those obtained in Scenario I. This result implies that a model trained by a file can predict the frames of a different file, surprisingly without the first part of the test data file. In Fig. 14, the best (HH_2) and worst (CC_2) cases can be found at $T = 0$, and then the performance change can be examined at other T values ($T = 4, 9$, and 14). It was found that the performance of HH_2 was always high (even at T of zero), being improved only slightly as T increased. In contrast, CC_2 showed a significant improvement as T increased, and was even better than HH_2 when $T = 14$.

3) Evaluation on good/bad cases over various time-steps. Fig. 15 shows the change in prediction performance over time-frames with different T ($\in \{0, 4, 9, 14\}$) for HH_2 and CC_2. For $T \leq 4$, the performance of HH_2 was still high (around 0.8) for all time-frames, while that of CC_2 was low (around 0.3). However, for $T \geq 9$, the prediction performance of CC_2 significantly rose for all values of T, e.g. 0.7 for $T = 9$ and 0.9 for $T = 14$. This result gives an insight into the properties of time frames in test data, as below. Starting from the beginning of a test file might not give the most accurate prediction performance. Instead,

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

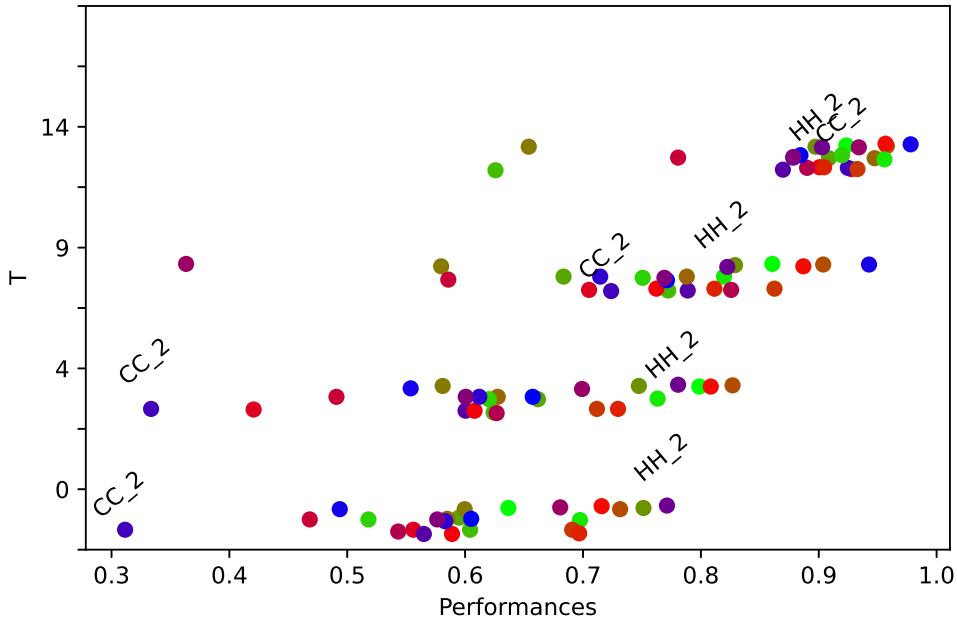

Figure 14: Prediction performances of Scenario II on the last frames for all test cases with different T values.

skipping several of the initial time frames might be beneficial for improving the prediction performance. This implies that the first time frames may not possess patterns consistent with those in the training data. Instead, intermediate or later time frames can share more patterns with the training data, which results in an improving predictive performance, a particular case being CC_2. Animations of HH_2 and CC_2 with $T = \{0, 4, 9, 14\}$ are available at Movie II: HH_2 and Movie II: CC_2, respectively.

Scenario III

1) Detailed experimental procedure. Fig. 16 illustrates the experimental procedure of Scenario III. Three files were combined from each environment (C, H, and L) to generate training data. For testing, the remaining file of each environment was used. Similar to Scenario II, T was used to skip the initial frames during testing. This scenario required a heavy computational load, and only one test file was able to be run for each of the six datasets.

2) Evaluation on the last frames. Fig. 17 shows the performances of the last frames of

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

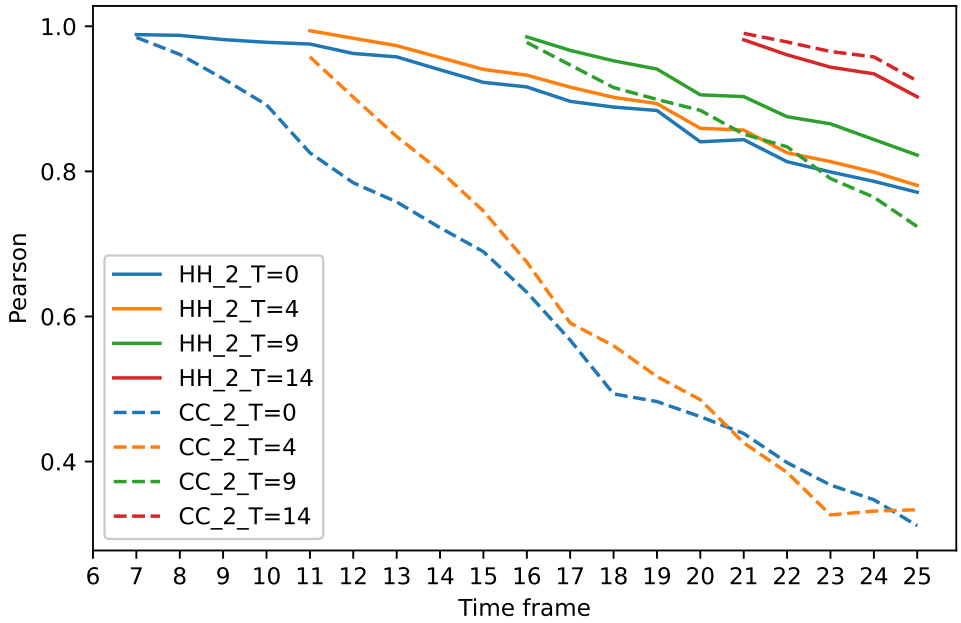

Figure 15: Prediction performances of Scenario II over various time-frames with different values of T.

the six datasets for all values of T (0, 4, 9 and 14). It can be seen that the performance results of Scenario III were similar to those of Scenario II. This indicates that a model can be trained using a collection of different environments to achieve a performance similar to that obtained by a model trained using only a single environment.

3) Evaluation on good/bad cases over various time-steps. The best and worst cases for $T = 0$ were HH_0 and CH_0, respectively. Fig. 18 shows the variation in the prediction performances over the different time frames of HH_0 and CH_0 when different T (0, 4, 9 and 14) was used. The prediction performance of CH_0 (worst case) was improved significantly from $T = 0$ to $T \geq 9$, which is consistent with the results of Scenario II. Overall it is clear that selecting an appropriate T value is to ensure an accurate prediction. Animations of HH_0 and CH_0 with $T = \{0, 4, 9, 14\}$ are available at Movie III: CH_0 and Movie III: HH_0, respectively.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

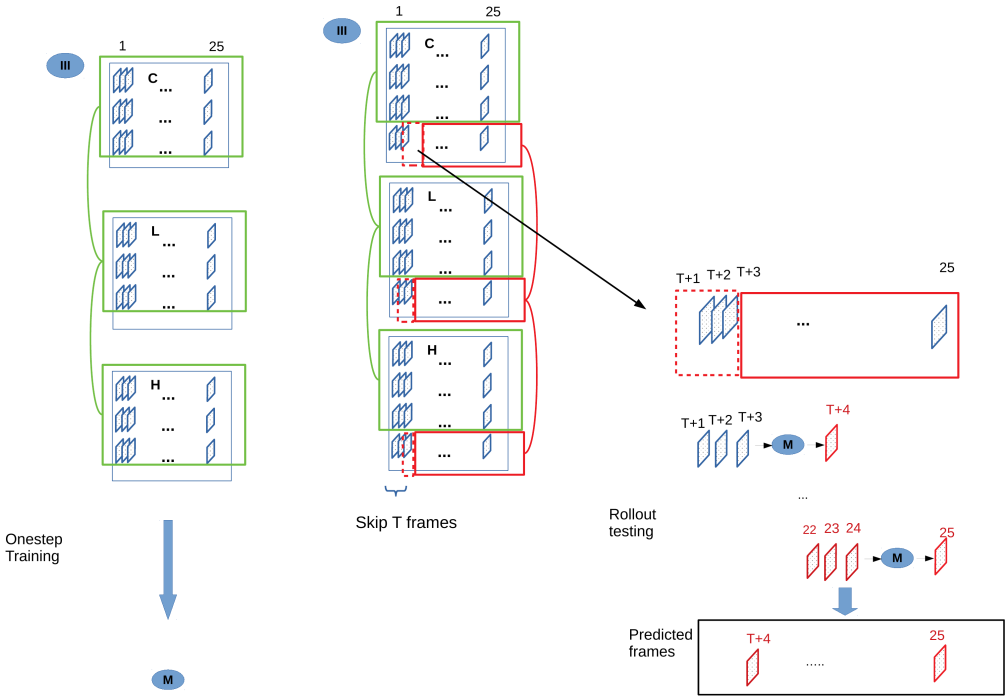

Figure 16: Training and testing in Scenario III.

Conclusions

The possibility of using a graph neural network-based simulator (GNS) to simulate the time-series self-assembly of biological cells was investigated, and the results obtained under three possible scenarios (See Fig. 6) were presented:

– **Scenario I: later time frames are predicted from earlier time frames in the same file.**

– **Scenario II: time frames are predicted from (all) time frames of other files under the same environment.**

– **Scenario III: time frames are predicted from (all) time frames of other files including those under different environments.**

From our experiments, the followings are the conclusions for the three scenarios:

– **Scenario I:** High prediction performances were achieved, indicating that later time frames can be predicted using earlier time frames of the same file.

– **Scenario II:** When T (the parameter to skip the earlier time frames in a test file) is small,

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

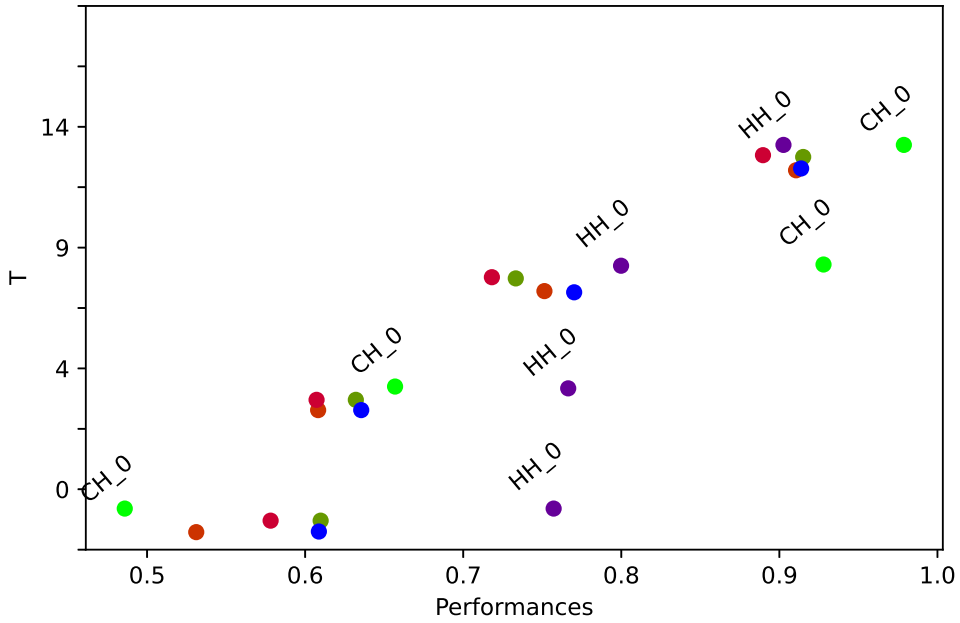

Figure 17: Prediction performances of Scenario III on the last frames of test cases.

the prediction performance was unfavourable. However upon increasing T, higher predictive performances were obtained, which were equivalent to those obtained in Scenario I. These results indicate that later time frames can be predicted by using the other files with the same properties as the test file. At the same time, it was concluded that the choice of T is important.

– **Scenario III:** Similar results to Scenario II were obtained, indicating that later time frames of a file can be predicted using the other files of different environments from the test file, if those files include the same property as the test file. This result also indicates that choosing T is important.

From these results, it can be clearly concluded that the use of a GNS is promising. In particular, from the results of the three scenarios, it was clearly shown that the GNS achieved highly favorable performances for later time frames of the test data, which supports the above conclusion more convincing. In more generaly, these results imply that a GNS would be useful to precisely simulate the self-assembly of cells and artificial ECM, i.e. the process of tissue (organ) regeneration. In particular, these results suggest that the GNS trained by

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

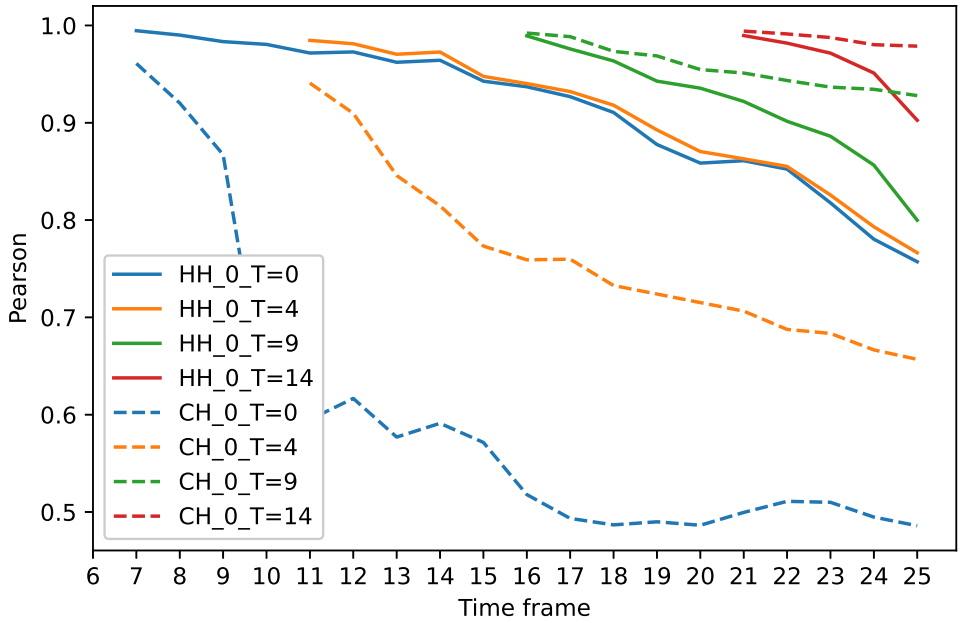

Figure 18: Prediction performances of Scenario III over various time frames with different values of T.

only 24 hour data could predict the cell types obtained after the differentiation of more than three weeks. This means that a GNS can reduce the number of experiments required to determine the possible conditions for generating particular cells with a certain 3D structure. Eventually, the GNS will accelerate the development of new materials for tissue regeneration and render tissue regeneration research more efficient. Finally again we emphasize that this is the first work of applying the GNS to predicting the cell differentiation types.

In order to improve the prediction performance of the GNS, more high-quality data with a larger number of time frames and smaller time steps are required. As an alternative to the GNS we used, the design of a new graph network model dedicated to biological systems would be interesting future work.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Author Contributions

The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

This work was in part supported by JST-CREST #JPMJCR21N7 (C.Y.), JSPS KAKENHI #22H02133 (C.Y.) #19H04169 (H.M.), #20F20809 (H.M.), #21H05027 (H.M.) and #22H03645 (H.M.) and the NIMS Joint Research Hub Program (C.Y.).

This work was performed in part on the NIMS Molecular and Material Synthesis Platform. We thank Dr. M. Shobo, NIMS for her technical support on gene expression analysis.

References

(1) Principles of Tissue Engineering (Fifth Edition). 2020; `https://www.sciencedirect.com/science/article/pii/B9780128184226000897`.

(2) Baker, B. M.; Chen, C. S. Deconstructing the third dimension – how 3D culture microenvironments alter cellular cues. *Journal of Cell Science* **2012**, *125*, 3015–3024.

(3) Jensen, C.; Teng, Y. Is It Time to Start Transitioning From 2D to 3D Cell Culture? *Frontiers in Molecular Biosciences* **2020**, *7*.

(4) Imamura, Y.; Mukohara, T.; Shimono, Y.; Funakoshi, Y.; Chayahara, N.; Toyoda, M.; Kiyota, N.; Takao, S.; Kono, S.; Nakatsura, T.; Minami, H. Comparison of 2D- and

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

3D-culture models as drug-testing platforms in breast cancer. *Oncol. Rep.* **2015**, *33*, 1837–1843.

(5) Diekjürgen, D.; Grainger, D. W. Polysaccharide matrices used in 3D in vitro cell culture systems. *Biomaterials* **2017**, *141*, 96–115.

(6) Yang, X.; Lu, Z.; Wu, H.; Li, W.; Zheng, L.; Zhao, J. Collagen-alginate as bioink for three-dimensional (3D) cell printing based cartilage tissue engineering. *Materials Science and Engineering: C* **2018**, *83*, 195–201.

(7) Melissaridou, S.; Wiechec, E.; Magan, M.; Jain, M. V.; Chung, M. K.; Farnebo, L.; Roberg, K. The effect of 2D and 3D cell cultures on treatment response, EMT profile and stem cell features in head and neck cancer. *Cancer Cell International 19*, 16.

(8) Bolognin, S. et al. 3D Cultures of Parkinson's Disease-Specific Dopaminergic Neurons for High Content Phenotyping and Drug Testing. *Advanced Science* **2019**, *6*, 1800927.

(9) Matsumura, K.; Rajan, R. Oxidized Polysaccharides as Green and Sustainable Biomaterials. *Current Organic Chemistry* **2021**, *25*, 1483–1496.

(10) Shi, W.; Sun, M.; Hu, X.; Ren, B.; Cheng, J.; Li, C.; Duan, X.; Fu, X.; Zhang, J.; Chen, H.; Ao, Y. Structurally and Functionally Optimized Silk-Fibroin–Gelatin Scaffold Using 3D Printing to Repair Cartilage Injury In Vitro and In Vivo. *Advanced Materials* **2017**, *29*, 1701089.

(11) Marchini, A.; Gelain, F. Synthetic scaffolds for 3D cell cultures and organoids: applications in regenerative medicine. *Critical Reviews in Biotechnology* **2022**, *42*, 468–486, PMID: 34187261.

(12) Song, Y.; Zhang, Y.; Qu, Q.; Zhang, X.; Lu, T.; Xu, J.; Ma, W.; Zhu, M.; Huang, C.; Xiong, R. Biomaterials based on hyaluronic acid, collagen and peptides for three-

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

dimensional cell culture and their application in stem cell differentiation. *International Journal of Biological Macromolecules* **2023**, *226*, 14–36.

(13) Spicer, C. D. Hydrogel scaffolds for tissue engineering: the importance of polymer choice. *Polym. Chem.* **2020**, *11*, 184–219.

(14) Mantha, S.; Pillai, S.; Khayambashi, P.; Upadhyay, A.; Zhang, Y.; Tao, O.; Pham, H. M.; Tran, S. D. Smart Hydrogels in Tissue Engineering and Regenerative Medicine. *Materials* **2019**, *12*.

(15) Chen, Y.; Dong, X.; Shafiq, M.; Myles, G.; Radacsi, N.; Mo, X. Recent Advancements on Three-Dimensional Electrospun Nanofiber Scaffolds for Tissue Engineering. *Advanced Fiber Materials* **2022**, *4*, 959–986.

(16) Niklason, L. E.; Gao, J.; Abbott, W. M.; Hirschi, K. K.; Houser, S.; Marini, R.; Langer, R. Functional Arteries Grown in Vitro. *Science* **1999**, *284*, 489–493.

(17) Pomahač, B.; Svensjö, T.; Yao, F.; Brown, H.; Eriksson, E. Tissue Engineering of Skin. *Critical Reviews in Oral Biology & Medicine* **1998**, *9*, 333–344, PMID: 9715370.

(18) Ma, P. X.; Langer, R. Morphology and mechanical function of long-term in vitro engineered cartilage. *Journal of Biomedical Materials Research* **1999**, *44*, 217–221.

(19) Mendes, A. C.; Baran, E. T.; Reis, R. L.; Azevedo, H. S. Self-assembly in nature: using the principles of nature to create complex nanobiomaterials. *WIREs Nanomedicine and Nanobiotechnology* **2013**, *5*, 582–612.

(20) Yoshikawa, C.; Hoshiba, T.; Sakakibara, K.; Tsujii, Y. Flocculation of Cells by Cellulose Nanofibers Modified with Concentrated Polymer Brushes. *ACS Applied Nano Materials* **2018**, *1*, 1450–1455.

(21) Yoshikawa, C.; Sakakibara, K.; Nonsuwan, P.; Shobo, M.; Yuan, X.; Matsumura, K. Cellular Flocculation Driven by Concentrated Polymer Brush-Modified Cellulose

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Nanofibers with Different Surface Charges. *Biomacromolecules* **2022**, *23*, 3186–3197, PMID: 35852304.

(22) Nonsuwan, P.; Nishijima, N.; Sakakibara, K.; Nakaji-Hirabayashi, T.; Yoshikawa, C. Concentrated polymer brush-modified cellulose nanofibers promote chondrogenic differentiation of human mesenchymal stem cells by controlling self-assembly. *J. Mater. Chem. B* **2022**, *10*, 2444–2453.

(23) Yuan, X.; Nonsuwan, P.; Shobo, M.; Rajan, R.; Yamazaki, T.; Sakakibara, K.; Matsumura, K.; Yoshikawa, C. Cellular Flocculation Using Concentrated Polymer Brush-Modified Cellulose Nanofibers with Different Fiber Lengths. *Biomacromolecules* **2022**, *23*, 1101–1111.

(24) Jorfi, M.; Foster, E. J. Recent advances in nanocellulose for biomedical applications. *Journal of Applied Polymer Science* **2015**, *132*.

(25) Hickey, R. J.; Pelling, A. E. Cellulose Biomaterials for Tissue Engineering. *Frontiers in Bioengineering and Biotechnology* **2019**, *7*.

(26) Tiwari, S.; Patil, R.; Bahadur, P. Polysaccharide Based Scaffolds for Soft Tissue Engineering Applications. *Polymers* **2019**, *11*.

(27) Khalil, H. P. S. A.; Jummaat, F.; Yahya, E. B.; Olaiya, N. G.; Adnan, A. S.; Abdat, M.; N. A. M., N.; Halim, A. S.; Kumar, U. S. U.; Bairwan, R.; Suriani, A. B. A Review on Micro- to Nanocellulose Biopolymer Scaffold Forming for Tissue Engineering Applications. *Polymers* **2020**, *12*.

(28) Tsujii, Y.; Ohno, K.; Yamamoto, S.; Goto, A.; Fukuda, T. In *Surface-Initiated Polymerization I*; Jordan, R., Ed.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2006; pp 1–45.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

(29) Yamamoto, S.; Ejaz, M.; Tsujii, Y.; Matsumoto, M.; Fukuda, T. Surface Interaction Forces of Well-Defined, High-Density Polymer Brushes Studied by Atomic Force Microscopy. 1. Effect of Chain Length. *Macromolecules* **2000**, *33*, 5602–5607.

(30) Yamamoto, S.; Ejaz, M.; Tsujii, Y.; Fukuda, T. Surface Interaction Forces of Well-Defined, High-Density Polymer Brushes Studied by Atomic Force Microscopy. 2. Effect of Graft Density. *Macromolecules* **2000**, *33*, 5608–5612.

(31) Tsujii, Y.; Nomura, A.; Okayasu, K.; Gao, W.; Ohno, K.; Fukuda, T. AFM studies on microtribology of concentrated polymer brushes in solvents. *Journal of Physics: Conference Series* **2009**, *184*, 012031.

(32) Nomura, A.; Okayasu, K.; Ohno, K.; Fukuda, T.; Tsujii, Y. Lubrication Mechanism of Concentrated Polymer Brushes in Solvents: Effect of Solvent Quality and Thereby Swelling State. *Macromolecules* **2011**, *44*, 5013–5019.

(33) Yoshikawa, C.; Goto, A.; Tsujii, Y.; Fukuda, T.; Kimura, T.; Yamamoto, K.; Kishida, A. Protein Repellency of Well-Defined, Concentrated Poly(2-hydroxyethyl methacrylate) Brushes by the Size-Exclusion Effect. *Macromolecules* **2006**, *39*, 2284–2290.

(34) Yoshikawa, C.; Goto, A.; Tsujii, Y.; Ishizuka, N.; Nakanishi, K.; Fukuda, T. Surface interaction of well-defined, concentrated poly(2-hydroxyethyl methacrylate) brushes with proteins. *Journal of Polymer Science Part A: Polymer Chemistry* **2007**, *45*, 4795–4803.

(35) Jordan, M. I.; Mitchell, T. M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260.

(36) Rickert, C. A.; Lieleg, O. Machine learning approaches for biomolecular, biophysical, and biomaterials research. *Biophysics Reviews* **2022**, *3*, 021306.

(37) Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature Cell Biology* **2015**, *521*, 436–444, Funding Information: Acknowledgements The authors would like to thank the

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

Natural Sciences and Engineering Research Council of Canada, the Canadian Institute For Advanced Research (CIFAR), the National Science Foundation and Office of Naval Research for support. Y.L. and Y.B. are CIFAR fellows. Publisher Copyright: © 2015 Macmillan Publishers Limited. All rights reserved.

(38) Angermueller, C.; Pärnamaa, T.; Parts, L.; Stegle, O. Deep learning for computational biology. *Molecular Systems Biology* **2016**, *12*, 878.

(39) Nishimoto, S.; Tokuoka, Y.; Yamada, T. G.; Hiroi, N. F.; Funahashi, A. Predicting the future direction of cell movement with convolutional neural networks. *PLOS ONE* **2019**, *14*, 1–14.

(40) Xu, M.; Papageorgiou, D.; Abidi, S.; Dao, M.; Zhao, H.; Karniadakis, G. A deep convolutional neural network for classification of red blood cells in sickle cell anemia. *PLOS Computational Biology* **2017**, *13*, e1005746.

(41) Nagao, Y.; Sakamoto, M.; Chinen, T.; Okada, Y.; Takao, D. Robust classification of cell cycle phase and biological feature extraction by image-based deep learning. *Molecular Biology of the Cell* **2020**, *31*, 1346–1354, PMID: 32320349.

(42) Kegeles, E.; Naumov, A.; Karpulevich, E. A.; Volchkov, P.; Baranov, P. Convolutional Neural Networks Can Predict Retinal Differentiation in Retinal Organoids. *Frontiers in Cellular Neuroscience* **2020**, *14*, 171.

(43) Zhu, Y.; Huang, R.; Wu, Z.; Song, S.; Cheng, L.; Zhu, R. Deep learning-based predictive identification of neural stem cell differentiation. *Nature Communications* **2021**, *12*, 1–13.

(44) Kipf, T. N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. Proceedings of the 5th International Conference on Learning Representations. 2017.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

(45) Sanchez-Gonzalez, A.; Godwin, J.; Pfaff, T.; Ying, R.; Leskovec, J.; Battaglia, P. Learning to simulate complex physics with graph networks. International Conference on Machine Learning. 2020; pp 8459–8468.

(46) Benesty, J.; Chen, J.; Huang, Y.; Cohen, I. *Noise reduction in speech processing*; Springer, 2009; pp 1–4.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0

TOC Graphic

Some journals require a graphical entry for the Table of Contents. This should be laid out "print ready" so that the sizing of the text is correct.
Inside the `tocentry` environment, the font used is Helvetica 8 pt, as required by *Journal of the American Chemical Society*.
The surrounding frame is 9 cm by 3.5 cm, which is the maximum permitted for *Journal of the American Chemical Society* graphical table of content entries. The box will not resize if the content is too big: instead it will overflow the edge of the box.
This box and the associated title will always be printed on a separate page at the end of the document.

https://doi.org/10.26434/chemrxiv-2023-9ll6s **ORCID:** https://orcid.org/0000-0002-6607-5617 Content not peer-reviewed by ChemRxiv. **License:** CC BY-NC-ND 4.0