

MODELING FLUID SIMULATION WITH DYNAMIC BOLTZMANN MACHINE

Kun Zhao, Takayuki Osogami

IBM Research - Tokyo
19-21, Nihonbashi Hakozaiki-cho, Chuo-ku
Tokyo, 274-0063, JAPAN

ABSTRACT

Fluid simulation requires a significant amount of computational resources because of the complexity of solving Navier-Stokes equations. In recent work [Ladický et al., 2015], a machine learning technique has been applied to only approximate, but to also accelerate, this complex and time-consuming computation. However, the prior work has not fully taken into account the fact that fluid dynamics is time-varying and involves dynamic features. In this work, we use a time-series machine learning technique, specifically the dynamic Boltzmann machine (DyBM) [Osogami et al., 2015], to approximate fluid simulations. We also propose a learning algorithm for DyBM to better learn and generate an initial part of the time-series. The experimental results suggest the efficiency and accuracy of our proposed techniques.

1 INTRODUCTION

Fluid simulation appears in various scientific simulations and computer graphic films. However, it requires a large amount of computational resources, because numerical solutions to Navier-Stokes equations are computationally expensive. To accelerate the simulation, many approximate algorithms have been proposed. Recently, machine learning techniques are also used to model this complex computation (e.g. [Ladický et al., 2015]). However, although fluid dynamics can be time-varying and dependent on historical states, only the latest state has been considered in the prior work.

In this work, we use a time-series machine learning technique called the dynamic Boltzmann machine (DyBM) [Osogami et al., 2015] to approximate fluid simulations. The DyBM can handle dynamic patterns and learn their dynamics via a learning rule that has the properties of spike-timing dependent plasticity, which is suitable for learning time-series of very high dimensions. We believe this feature makes the DyBM particularly suitable to model dynamic fluid simulations. However, the traditional DyBM does not necessarily perform well on an initial part of a time-series. To reduce the error in approximate simulation by a DyBM, we propose a learning framework for the DyBM to handle the initial part separately. As a result, our proposed technique can learn to quickly simulate the dynamics of fluid with a small error of approximation. We conducted experiments of learning fluid simulations and compared the proposed method against the one with a conventional static machine learning model. The results demonstrate the efficiency and accuracy of our proposed method.

2 PROPOSED METHOD

2.1 Review of Dynamic Boltzmann Machine

A DyBM is derived from a Boltzmann machine, but involves additional memory units. A neuron is connected to another via a synapse with conduction delay, d . A first-in first-out (FIFO) queue causes this conduction delay and stores the values of the pre-synaptic neuron for the last $d - 1$ units of time. The value of the pre-synaptic neuron reaches the synapse after the conduction delay. Moreover, the DyBM aggregates information about the spikes in the past in the form of neural eligibility traces and synaptic eligibility traces,

which are stored in the memory units. In the training process, the parameters (weights associated with FIFO queues and eligibility traces, as well as biases associated with neurons) are optimized to decrease the energy of the DyBM for training data of the time-series. After iterations of training, the DyBM learns a generative model of the dynamic patterns. In the generation process, preceding $d - 1$ steps of the time-series (in the FIFO queues) and the values stored in the eligibility traces are used as inputs to generate the next pattern, which is then used to update the FIFO queues and eligibility traces. For more details, see [Osogami and Otsuka, 2015].

2.2 Learning Framework

A fluid simulation is performed from given initial conditions and continues for a period specified by a user. A DyBM can learn to predict the next state of the fluid given its preceding states. In this work, we ignore eligibility traces and use only FIFO queues as the memory units of a DyBM. The DyBM then predicts the next state on the basis of the preceding states for the last $d - 1$ time-steps. However, in standard fluid simulations, only the initial state of the fluid is given, and its following states need to be predicted solely from the initial state. This means that the traditional DyBM cannot be used directly for a simulation from the initial state. To enable a simulation with a DyBM from the initial state of fluid, we propose a framework for carefully handling the initial part of the time-series in the learning and generation processes (Figure 1).

In the learning process, we train multiple DyBMs with different delays. The delays are integers with a range from 2 to an optimized value of m , where m is the largest delay needed to learn the dynamics of fluid under consideration. The DyBMs with a delay of 2 to $m - 1$ are used to handle the initial time-series, and the DyBM with a delay of m is used to handle the rest of the time-series.

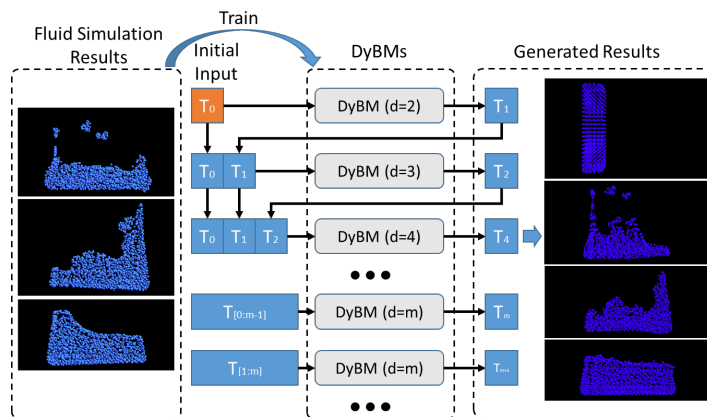


Figure 1: Generation process with trained DyBMs. Only initial state T_0 of fluid is given as a cue to generate dynamics of the fluid.

In the generation process, these trained DyBMs are used to generate a time-series of the states of the fluid given its initial state T_0 only (see the top part of Figure 1). This initial state contains the start locations of particles that form the fluid. From the initial state T_0 , the DyBM with *delay* = 2 is used to predict the next state T_1 . This predicted T_1 and the initial state T_0 are then used as the input to the DyBM with *delay* = 3 to predict T_2 . This process continues with an increased delay (Figure 1). Note that, after T_m is generated, we keep using the DyBM with *delay* = m to generate the rest of the time-series of the fluid state.

3 EXPERIMENTS

In our experiments, we first created training data by simulating the dynamics of fluid in a container by a standard method of smoothed-particle hydrodynamics (SPH) [Lucy, 1977] from the initial positions of particles shown in the left-most panel of Figure 2. The container was sized as 20.0, 20.0, and 10.0 in x , y , and

z axis, respectively. This data was then used to train DyBMs with different delays. The maximum delay used in this experiment is $m = 3$. The simulation data consists of the positions of 1,656 particles (4,968 variables for x, y, z coordinates) for 400 time-steps. We conducted this experiment on a notebook PC with an Intel Core i5 CPU (1.6 GHz) and 8 GB of system memory.

In this experiment, the simulation with SPH took about 210 seconds. On the other hand, the DyBM only required about 25 seconds to approximately simulate the dynamics of the fluid for the entire 400 steps. The root mean square error (RMSE) of the approximately simulated positions of the particles was 0.044. As shown in Figure 3, the approximate simulation with the DyBM gives the appearance of fluid flowing in a container.

As a comparison, we also conducted an experiment using the standard linear regression model instead of a DyBM. The linear regression model uses only the preceding state of particles to predict the next state of those particles without taking into account the dynamic features, which were used by the DyBM. The approximate simulation with the linear regression model resulted in the RMSE of 2.21×10^{57} . This huge error is because the linear regression only considers the static feature and does not have the capability to model the whole simulation. Particles diffused very roughly in the simulation modeled by linear regression, due to the learning's inability to converge into stable parameters.

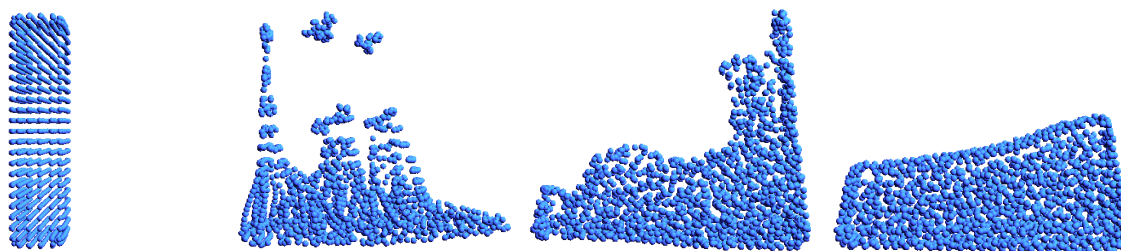


Figure 2: Approximate fluid simulation by proposed method. Left-most panel shows initial state of particles. Other panels show state of particles at 20th, 50th, and 155th time-steps, which are supposed to be representative for this simulation..

4 CONCLUSIONS

In this work, we proposed a machine learning-based dynamic fluid approximation method by using the dynamic Boltzmann machine (DyBM). To enhance the utility, we also proposed a learning algorithm for DyBM to better learn and generate an initial part of the time-series. We conducted an experiment to model a SPH-based fluid simulation. The experimental results show that our proposed method can get a much faster speed to generate the simulation while maintaining a much better accuracy than a traditional static machine learning model.

ACKNOWLEDGMENTS

This work was supported by JST CREST Grant Number JPMJCR1304, Japan.

REFERENCES

- Ladický, L., Jeong, S. H., Solenthaler, B., Pollefeys, M., and Gross, M. 2015. Data-driven fluid simulations using regression forests. *ACM Transactions on Graphics* **34**, 6, Article 199 (October 2015), 9 pages.
- Osogami, T. and Otsuka, M. 2015. Seven neurons memorizing sequences of alphabetical images via spike-timing dependent plasticity. *Scientific Reports* **5**, Article 14149.
- Lucy, L. B. 1977. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal* **82**, 1013–1024.