Heuristic Modeling of Deformable Object using Node-based Structure with Mass-spring System

Fakhrul Syakirin Omar, Md. Nazrul Islam, and Habibollah Haron

Abstract—Modeling of deformable objects can be found in practically every place and usage, from cloths and soft materials in virtual environment to skin and internal organs found in surgery simulation. With the model, actual object can be described in its physical entirety thus can be tested, destroyed and rebuild easily which reduces the experimentation cost significantly. This paper presents a simple approach to solve object reaction for model which adapts mass-spring technique to emulate elastic behavior in the solid. Iterative algorithm is used to achieve force equilibrium throughout model structure that is layered based on proximity from loading area. Stability of the process is assured by terminating condition included in the algorithm.

Index Terms—deformable object, heuristic mechanic, mass-spring system, mesh deformation, physically based modeling

1 Introduction

DEFORMABLE object modeling is different from the solid type by the continuous smooth reaction rather than instantaneous change occurred to the material. The external force thus will be distributed along the object infrastructure and may affect only a small effective area of the solid part, not at a whole object per se. Two major framework classes are typically being implemented to solve the problem: mass-spring-damper network and finite element method. In this paper, a variant of the former class is introduced with exclusion of solving ordinary differential equations requirement attributed to the model. Damper and mass components are removed from link connecting point nodes, leaving only spring to emulate elasticity of the material.

The focus of the algorithm is procedural simplicity and lightweight processing, made possible by layer segmentation starting from loading area and incremental loading force distribution throughout the network. Output accuracy and processing time which are directly compromised from the new configuration are used as benchmark to compare between models available. The ideal system will be the highest accuracy reached with lowest time consumed to complete the calculation.

The remaining of this paper is arranged as follows: In section "Previous Works in Deformable Object Modeling", existing literature related to the topic discussed in this paper is presented and followed with section "Research Methodology" which clarifies the flow and steps taken in order to develop the algorithm and complete the research. Next section, "Heuristic Deformable Object Modeling" contains the proposed model details including the principle and equation used. "Implementation and Experiment" section describes the setup and environment to test proposed model performance and functionality in terms of development method and testing technique followed with "Result and Analysis" which compiles and discusses the output of the experiment. Lastly, "Conclusion and Future Work" section wraps up the paper based on its objective and results and suggests possible steps to enhance and improve the algorithm.

- *Fakhrul Syakirin Omar is with the Department of Computer Science, Universiti Teknologi Malaysia, Malaysia, Johore 81310. E-mail: fsyakirin2@live.utm.my*
- *Md. Nazrul Islam is with the Department of Computer Science, Universiti Teknologi Malaysia, Malaysia, Johore 81310. E-mail: mdnazrul@utm.my*
- *Habibollah Haron is with the Department of Computer Science, Universiti Teknologi Malaysia, Malaysia, Johore 81310. E-mail: habib@utm.my*

2 Previous Works in Deformable Object Modeling

Two popular frameworks in modeling deformable object are finite element method (FEM) and mass-spring network. The differences between these types are concerned with highest accuracy can be achieved, processing time taken for the calculation and simplicity to implement the model. FEM which having its background from civil engineering can reach high precision with increasing model parameter complexity but time taken to solve partial differential equation (PDE) involved in the model makes this style unpopular for system requiring real-time performance. Excluding from time constraint, FEM has been found its way to simulate deformable object in various fields such as cloth modeling [1], [2] and food product [3]. Improvement to FEM is available such as PDE speedy calculation method, and FEM variants for example finite volume method [4] and boundary element method (BEM) [5] that only utilizes mesh at object's outer surface is introduced as possible replacement to the technique.

The other main modeling method for deformable object is mass-spring network (MSN) which utilizes spring and damper elements, assembled in certain arrangement to emulate material elasticity. Its simplicity in terms of implementation and lightweight calculation are the attractive point as a model though it can never be accurate. Thus, MSN model usage is abundant where execution speed is prioritized more than solution accuracy to achieve realistic reaction to the user in virtual reality field such as in surgery simulation [6] and rocking motion [7]. This method has been developed further for specific purpose such as in [8], [9], [10] or solving improvement in [11], [12].

Various alternative approaches are available such as impulse- or particle-based method [13], [14], [15], [16] which represents a solid with a collection of particles or point cloud interacting to force via collision analysis, usage of pre-computed or collection of data to be recorded and retrieved as response [17], geometrical-based model [18], physicochemically inspired reaction-diffusion process [19] to enable non-linear elasticity, and statistically-based model [20] which compares input variation to the reference structure. Furthermore, coupling of these methods has been reported to optimize specific performance factor such as [21], [22].

The algorithm presented in this paper aims to further simplify implementation of MSN model by directly repositioning nodes in object structure by force accumulation at each node segmented by layers which enables node removal emulating cutting operation and parallel calculation.

3 Research Methodology

There are four stages involved in this research namely the data definition, development of deformable object modeling algorithm, algorithm implementation, and followed with testing and analysis of the result. First phase, data definition is identification and collection for the correct input and output of the algorithm. Deformable object is presented as a block of springs connected in a certain manner creating a network which can stretch and compress as response to external force. Sample input is generated manually with its material properties such as spring hardness is arbitrarily set. The input is then manipulated by the developed algorithm and finally the deformed block of springs will be produced.

In second phase, the algorithm is developed based on logical flow of force expansion observed when a soft cushion is being crushed from a heavy object. The vector distribution is then converted into a set of rules to generalize the situation when generic object shape and material is involved. Then the steps are implemented as C program to produce a working implementation. In final stage, the algorithm is tested by simulating external force against the sample model and observing model response with the increasing of force magnitude, which acts as analysis for the result.

4 Heuristic Deformable Object Modeling

Preparation of solid material model in terms of node structure and algorithm used in the modeling are explained in this section. These are the three steps involved namely node arrangement, node network layering, and force calculation and distribution.

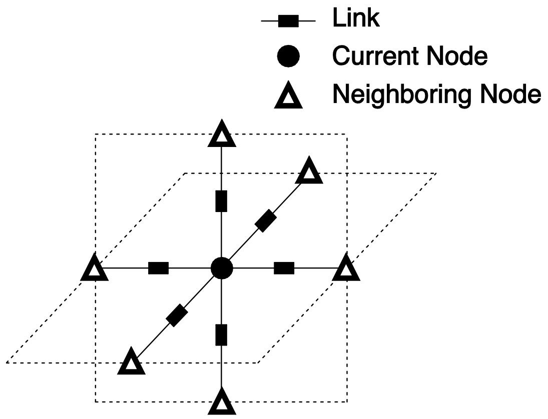


Fig. 1. Link interconnection between nodes in proposed network. Only a single spring is embedded for each link.

4.1 Node Arrangement

The nodes in the network are arranged in rectangular structure, with 6 links attached to each node for connection to other neighboring nodes. Resulting outline formed by node network would be referring to actual shape of any object being tested, which is sliced based on predefined grid size to set each node position. All node is connected to its neighborhood via links which containing a single spring, set to behave linearly according to acting force. Diagram in Fig. 1 helps visualize this node interconnection system.

4.2 Layering in Node Network

Once the loading area has been identified, nodes included in the area are marked as active or current nodes. Its immediate neighboring node directly connected to this is called layer 1 node as the first layer to be repositioned due to the acting force. This single node stripe is defined as a layer though they have no any additional connection between them due to the grouping. In Fig. 2 below, member for each respective layer is shown to be connected with intra-layer link to provide clearer view of its member positioning around the current node.

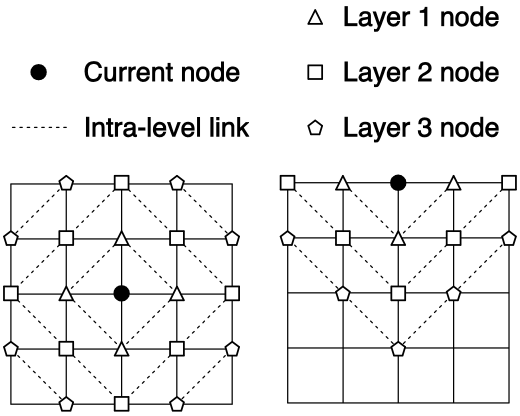


Fig. 2. Layering in node network starting from current node using top and cross-section views.

Subsequent expansion by repeatedly searching in

```
Start at layer immediately surrounding the loading area
Until all nodes in current layer reached equilibrium
  For every node in current layer
    Until acting force at current node is stabilized
      Accumulate acting force at current node
      Move node in effective force direction axis
    End
  End
  If position change disrupts neighboring node balance
    Raise need to check next layer
  End
End
```

Fig. 3. Steps to distribute resulting force from loading area to the surrounding node network.

outward direction for untouched node is done sequentially from previously found layer until the effect of acting force is small enough to disrupt the force balance at current layer node. Until then, force is dispersed per layer basis starting from the loading area.

4.3 Force Calculation and Distribution

Standard spring equation F=-kx has been used for all embedded spring elements where F is the resulting force from the spring displacement, x is the displacement amount when the spring is stretched or compressed and k is the constant defining the elasticity or strength of the spring. Its linearity provides material response predictability and simplifies testing implementation part thus is chosen to govern the spring and consequently the model behavior.

The calculation is based on effective acting force at every node contained in current layer. With the exception at loading area which actually starts the imbalance in node effective force, the pulling or pushing force contributed from every connected link at each node is accumulated to identify the direction for adjustment movement of the node in order to stabilize its force equilibrium. Detailed steps taken for this are given above in Fig. 3:

As for the loading area part, different approach is taken to introduce force imbalance in node network as reaction to the loading event. The whole loading area is regarded as a single hard surface patch to reflect the shape of contact surface and the force contribution due to links attached to the patch border will be regarded as the initial material response. Except from this, as in force distribution in subsequent layers, the surface patch repositioning is performed until cumulative link force is equal to the loading force. Fig. 4 shows the loading node and its encompassing area that is said to be assumed as stiff to be moved simultaneously.

5 Implementation And Experiment

For the implementation of the algorithm, GNU GCC compiler has been used to build the processing part while Gnuplot program enables the output model to be visualized interactively. A PC running Windows XP 64-

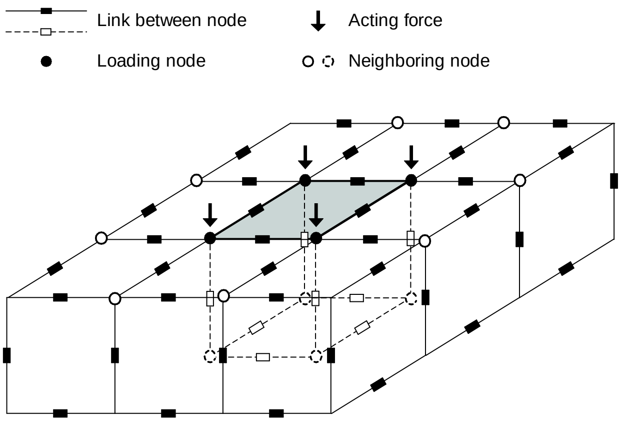


Fig. 4. Working node network. Loading area is colored with gray shade which is imposed with external acting force depicted with arrows.

bit with AMD Athlon X4 3.0GHz and 4GB RAM which the program utilized only a single core is used for the platform.

The setup being used is a solid cuboid with its properties in terms of node count and spring hardness is kept constant throughout the experiment. The model is set to be floating in space; that is not put on surface that can disrupt changes at any model face and edges. As for the testing, external force being exerted to the model is gradually increased and its effect to the model shape is observed visually.

6 Results and Analysis

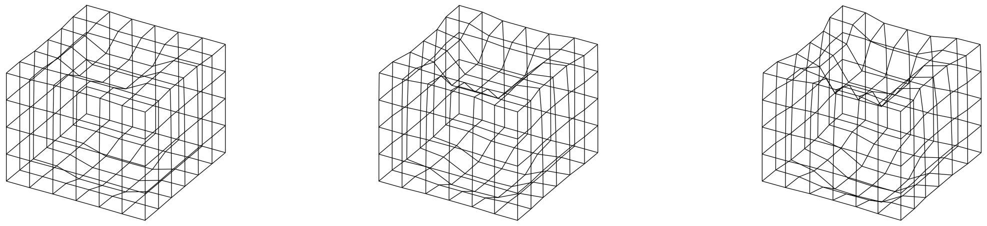
Resulting deformation to the object from various configuration of external force can be observed from wireframe visualization of the model, compiled in Fig. 5 and 6. The increasing force magnitude exerted results in higher deformation degree in the model when the figure set is viewed from (a) to (f).

Volume preservation of the object is maintained in the model when examining the collapse and bulging at its edge and bottom as the loading area is pushing into the object in Fig. 5. The processing time taken seems to be dependent to the force direction complexity, from simple straight downward in z-axis in Fig. 5 to include all axes in Fig. 6. Apparently the situation arises from the need to compute 2 additional normal vectors to address changes occurring in each axis.

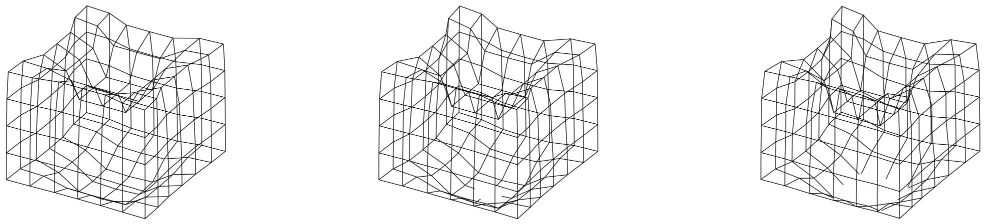
7 Conclusion and Future Work

This paper presented a simple deformable object modeling method using mass-spring network by introducing an iterative approach for applying external force to the object. The force is applied in steps to ensure smooth transformation in node positioning, and in layers for equal distribution throughout the network.

In order to improve the processing speed, obvious solution is to enable parallel execution at each node that

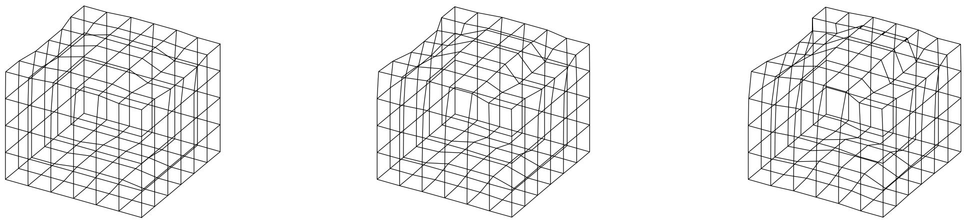


(a) Magnitude=2, Time=0.188s (b) Magnitude=4, Time=0.111s (c) Magnitude=6, Time=0.125s

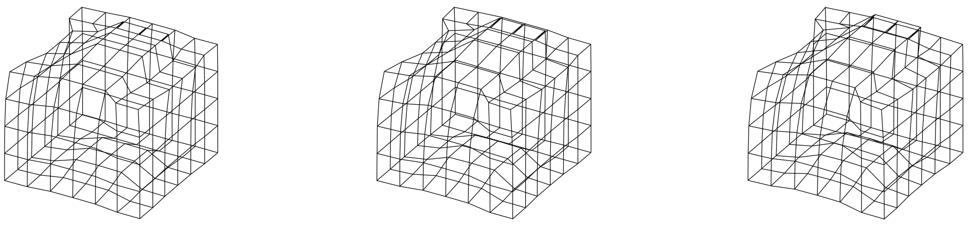


(d) Magnitude=8, Time=0.141s (e) Magnitude=10, Time=0.203s (f) Magnitude=12, Time=0.172s

Fig. 5. Deformation of solid model due to straight downward force with increasing magnitude. Execution time taken is given for comparison purpose.



(a) Magnitude=1, Time=0.329s (b) Magnitude=2, Time=0.266s (c) Magnitude=3, Time=0.281s



(d) Magnitude=4, Time=0.312s (e) Magnitude=5, Time=0.329s (f) Magnitude=6, Time=0.312s

Fig. 6. Deformation of solid model due to diagonal upward force with increasing magnitude. Execution time taken is given for comparison purpose.

does not connect to each other to still maintain validity of the output. Subroutine to automatically determine force step size depending on solid spring constant and mesh size is also useful to balance accuracy and speed of the algorithm. Finally, further testing regarding to suturing and cutting operation to the model could be done to examine behavior of the model due to internal stress.

ACKNOWLEDGMENT

This paper is supported by Zamalah student funding and Research University Grant project vote number 4D057 from Universiti Teknologi Malaysia and Malaysian Ministry of Higher Education.

REFERENCES

[1] S. Sumengen, M.T. Eren, S. Yesilyurt, S. Balcisoy, A Multi-Resolution Mesh Representation For Deformable Objects In Collaborative Virtual Environments. *Computer Vision and Computer Graphics. Theory and Applications*, in: J. Braz, A. Ranchordas, H.J. Araújo, J.M. Pereira (Eds.), vol. 21, Springer Berlin Heidelberg, pp. 75-87, 2009.

[2] R. Wang, Y. Liu, X. Luo, Y. Li, S. Ji, A Finite-Element Mechanical Contact Model Based On Mindlin–Reissner Shell Theory For A Three-Dimensional Human Body And Garment, *Journal of Computational and Applied Mathematics*, vol. 236, pp 867-877, 2011.

[3] Z. Wang, S. Hirai, Modeling And Estimation Of Rheological Properties Of Food Products For Manufacturing Simulations, *Journal of Food Engineering*, vol. 102, pp 136-144, 2011.

[4] J. Sewall, N. Galoppo, G. Tsankov, M. Lin, Visual Simulation Of Shockwaves, *Graphical Models*, vol. 71, pp 126-138, 2009.

[5] Y.-M. Tang, K.-C. Hui, Human Foot Modeling Towards Footwear Design, *Computer-Aided Design*, vol. 43, pp 1841-1848, 2011.

[6] P. Patete, M.I. Iacono, M.F. Spadea, G. Trecate, D. Vergnaghi, L.T. Mainardi, G. Baroni, A Multi-Tissue Mass-Spring Model For Computer Assisted Breast Surgery, Medical Engineering & Physics. doi:10.1016/j.medengphy.2012.03.008

[7] M.N. Chatzis, A.W. Smyth, Modeling Of The 3D Rocking Problem, *International Journal of Non-Linear Mechanics*, vol. 47, pp 85-98, 2012.

[8] S. Arnab, V. Raja, A Deformable Surface Model With Volume Preserving Springs. *Articulated Motion And Deformable Objects*, in: F. Perales, R. Fisher (Eds.), vol. 5098, Springer Berlin / Heidelberg, pp. 259-268, 2008.

[9] J. Das, N. Sarkar, Autonomous Shape Control Of A Deformable Object By Multiple Manipulators, *Journal of Intelligent & Robotic Systems*, vol. 62, pp 3-27, 2011.

[10] V.B. Zordan, B. Celly, B. Chiu, P.C. DiLorenzo, Breathe Easy: Model And Control Of Human Respiration For Computer Animation, *Graphical Models*, vol. 68, pp 113-132, 2006.

[11] X. Hu, L. Wei, D. Li, A Modified Numerical Integration Method For Deformable Object Animation. *AsiaSim* 2007, in: J.-W. Park, T.-G. Kim, Y.-B. Kim (Eds.), vol. 5, Springer Berlin Heidelberg, pp. 375-383, 2007.

[12] S. Natsupakpong, M. Cenk Çavusoglu, Determination Of Elasticity Parameters In Lumped Element (Mass-Spring) Models Of Deformable Objects, *Graphical Models*, vol. 72, pp 61-73, 2010.

[13] R. Goldstein, G. Wainer, Impulse-Based Dynamic Simulation Of Deformable Biological Structures. *Transactions on Computational Systems Biology XIII*, in: C. Priami, R.-J. Back, I. Petre, E. de Vink (Eds.), vol. 6575, Springer Berlin / Heidelberg, pp. 39-60, 2011.

[14] B. Zhu, L. Gu, Z. Zhou, Particle-Based Deformable Modeling With Pre-Computed Surface Data In Real-Time Surgical Simulation. *Medical Imaging and Augmented Reality*, in: H. Liao, P. Edwards, X. Pan, Y. Fan, G.-Z. Yang (Eds.), vol. 6326, Springer Berlin / Heidelberg, pp. 503-512, 2010.

[15] A. Henriques, B. Wünsche, Improved Meshless Deformation Techniques For Plausible Interactive Soft Object Simulations. *Computer Vision and Computer Graphics. Theory and Applications,* in: J. Braz, A. Ranchordas, H.J. Araújo, J.M. Pereira (Eds.), vol. 21, Springer Berlin Heidelberg, pp. 88-101, 2009.

[16] K. Tagawa, K. Hirota, M. Hirose, Surface Contact Interaction With Dynamically Deformable Object Using Impulse-Based Approach. *Haptics: Perception, Devices and Scenarios*, in: M. Ferre (Ed.), vol. 5024, Springer Berlin / Heidelberg, pp. 675-680, 2008.

[17] K. Tagawa, K. Hirota, M. Hirose, A Study On Haptic Interaction And Simulation Of Motion And Deformation Of Elastic Object. Human Interface And The Management Of Information. *Methods, Techniques and Tools in Information Design*, in: M. Smith, G. Salvendy (Eds.), vol. 4557, Springer Berlin / Heidelberg, pp. 985-993, 2007.

[18] Q. Liu, E.C. Prakash, M.A. Srinivasan, Interactive Deformable Geometry Maps, *The Visual Computer*, vol. 23, pp. 119-131, 2007.

[19] Y. Zhong, B. Shirinzadeh, J. Smith, C. Gu, Soft Tissue Deformation With Reaction-Diffusion Process For Surgery Simulation, *Journal of Visual Languages & Computing*, vol. 23, pp. 1-12, 2012.

[20] P. Du, H. Ip, B. Hua, J. Feng, Using Surface Variability Characteristics For Segmentation Of Deformable 3D Objects With Application To Piecewise Statistical Deformable Model, *The Visual Computer*, vol. 28, pp 493-509, 2012.

[21] B. Zhu, L. Gu, A Hybrid Deformable Model For Real-Time Surgical Simulation, Computerized Medical Imaging and Graphics. doi:10.1016/j.compmedimag.2012.03.001

[22] U. Güdükbay, S. Bayraktar, Ç. Koca, B. Özgüç, Particle-Based Simulation Of The Interaction Between Fluid And Knitwear, Signal, Image and Video Processing, 1-8. doi:10.1007/s11760-012-0308-2

Fakhrul Syakirin Omar is a researcher in Operational Business Intelligence group in Universiti Teknologi Malaysia (UTM). He obtained his Bachelor degree in Microelectronics in 2009 at UTM and is currently pursuing his MSc at the same institution. His working experiences cover image processing and problem solving using soft computing techniques.

Md. Nazrul Islam is a senior lecturer at Faculty of Computing, Universiti Teknologi Malaysia. He received his PhD in Information Science from University of Fukui, Japan. His research focuses on path planning & motion planning algorithm for robotics manipulators, assistive robotics, human-robot interaction modeling, collision detection algorithm in deformable environment for medical robot and kinematics & dynamics of parallel manipulators.

Habibollah Haron is Professor and Head, Department of Computer Science, Faculty of Computing, Universiti Teknologi Malaysia. His research interests include operational research, soft computing and computer aided geometric design.