

DOI: <http://dx.doi.org/10.21123/bsj.2020.17.4.1294>

## Real-Time Cloth Simulation on Virtual Human Character Using Enhanced Position Based Dynamic Framework Technique

*Itimad Raheem Ali* <sup>\*1</sup>

*Mazin Abed Mohammed* <sup>2</sup>  
*Hoshang Kolivand* <sup>4</sup>

*Talib A. Al-Sharify* <sup>3</sup>

<sup>1,\*</sup>Department of MIS, Faculty of Business Informatics, University of Information Technology and Communications, Baghdad, Iraq.

<sup>2</sup>Department of Computer Science, College of Computer Science and Information Technology, University of Al-Anbar, Anbar, Iraq.

<sup>3</sup>Department of Communications Engineering, College of Engineering Sciences, Al-Rafidain University Collage, Baghdad, Iraq.

<sup>4</sup>Department of Computer Science, Faculty of Engineering and Technology, Liverpool John Moores University, L3 3AF, Liverpool, UK.

\*Corresponding author: [itimadra@uoitc.edu.iq](mailto:itimadra@uoitc.edu.iq)\*, [mazinalshujeary@uoanbar.edu.iq](mailto:mazinalshujeary@uoanbar.edu.iq), [talibalsharify1955@gmail.com](mailto:talibalsharify1955@gmail.com), [h.kolivand@ljmu.ac.uk](mailto:h.kolivand@ljmu.ac.uk).

\*ORCID ID: <https://orcid.org/0000-0002-9762-0534>\*, <http://orcid.org/0000-0001-9030-8102>, <https://orcid.org/0000-0002-3657-5944>, <https://orcid.org/0000-0001-5460-5679>.

Received 21/8/2019, Accepted 20/5/2020, Published 1/12/2020



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

### Abstract:

Cloth simulation and animation has been the topic of research since the mid-80's in the field of computer graphics. Enforcing incompressible is very important in real time simulation. Although, there are great achievements in this regard, it still suffers from unnecessary time consumption in certain steps that is common in real time applications. This research develops a real-time cloth simulator for a virtual human character (VHC) with wearable clothing. This research achieves success in cloth simulation on the VHC through enhancing the position-based dynamics (PBD) framework by computing a series of positional constraints which implement constant densities. Also, the self-collision and collision with moving capsules is implemented to achieve realistic behavior cloth modelled on animated characters. This is to enable comparable incompressibility and convergence to raised cosine deformation (RCD) function solvers. On implementation, this research achieves optimized collision between clothes, syncing of the animation with the cloth simulation and setting the properties of the cloth to get the best results possible. Therefore, a real-time cloth simulation, with believable output, on animated VHC is achieved. This research perceives our proposed method can serve as a completion to the game assets clothing pipeline.

**Keywords:** Cloth Simulation, Collision Constraint, Movement Force, Position Based Dynamics, Raised Cosine Deformation Functions.

### Introduction:

Computer graphics is, in recent time, extensively applied in education, entertainment, and even in the military. A number of research focus in the area of computer graphics is geared towards advancing computer graphics to serve other areas of computer science in need of models and animation. The earliest models to exist in computer graphics was created by Fabian et al. (1). He created a simulated naval aircraft model. One of such models trending in the graphics industry is cloth models. Cloth simulations play an important role in representing a computer-generated character in a

simulated world (2, 3, 4). Despite many years of research, cloth simulation and animation remain an important topic in the field of computer graphics and multimedia tools. As the application industry of graphics advances, the yearn for quality of the results widens. Usually, researchers try to achieve quality through employing cloth simulation methods such as physical, geometric, hybrid and data-driven methods. However, the challenge is in the trade-off between realism and efficiency. For applications which require off-line simulations, like in movies, efficiency can be compromised over

achieving a more realistic cloth simulation. The real-time applications such as virtual reality in computer games or the garment simulation industry, the reverse is the case (5). The latter is of interest in this paper.

The simulation of cloth in virtual environment is difficult and efforts to achieve realism have existed for a decade. Early efforts were in the practical recreation of the mechanical conduct of material. Its advances are towards reproduction of virtual articles of clothing on engineered characters. While modeling is advantageous to cloth recreation on enlivened virtual characters, virtual prototyping of piece of cloth model also benefits the industry. The mechanical models are more suitable for modelling the rigorous nature of cloth due to their root in the creation of deformable structures for mechanical buildings. Cloth presents great deal of difficulties because of the flexible nature of cloth material and the expected behavior of wrinkling, overlaying unto itself, and responding to wind direction. Therefore, for the graphics industry to achieve realism and effectiveness in cloth simulation, there is need for a mechanical deformation model that can create the mechanical properties of cloth material. However, due to the natural and exceptional deformable nature of cloth material, it is expected that the mechanical representation of a cloth must be sufficiently exact, that is, be able to reveal the nonlinearities and the enormous irregularities in a material. As a result, achieving utmost realism might prove challenging.

The fit of cloth to an avatar, is yet another significant problem in cloth simulation. Cloth fabric, which can take different material properties, should firmly wrap around a body. In literature, it takes techniques that can effectively distinguish the geometrical change that different fabric structure exhibits, and integrates into the mechanical model. As much of the computational complexity required in cloth simulation is as much needed in fitting the cloth to a body for quality of result. However, in attaining such realism, calculation speed and effectiveness of the cloth simulation model is substantially crucial in application of the end result.

Therefore, this paper aims at implementing real-time simulation of cloth meshes on game characters to aid cloth authoring. The contributions makes is in achieving modelling and animation of virtual human characters with realistic cloth movements in real-time using the position based dynamics (PBD) approach. Particularly, the focus is on synthesizing animation to virtual human character movements which can be made possible through PBD (6). The research utilized the open-

source library that is made available in (5). The research further analyzes state-of-the-art real-time cloth simulators such as Syflex, APEX cloth, and Havok cloth which provide cloth simulation solutions for games. The analysis is on the bases of the simulator's workflow; cloth authoring using 3D graphics packages (like Maya, 3DS max and Unity), and plug-in to a game development environment like unity developer environment(UDE), where simulation tests can be performed in real-time.

The rest of this paper is organized as follows: Section 2 reviews previous work. Presented in Section 3 is the proposed method and in Section 4 is the experimental result and discussions which include images and testing protocol for real-time and state of art cloth simulators. Finally, Section 5 concludes the study and outlines several possible recommendations for further research.

### Previous Work

Due to the growing need for realistic animations, cloth simulation has become an emerging topic of interest in the graphics industry. Since clothes do not consist of internal controllers its behavior may be passive to the surrounding environment (7). Physical based simulations are helpful in analysing and modelling the behaviour of clothes. In this work (8) the physical based simulation is approached from the perspective of low-dimensional linear subspaces like the principal component analysis, which represent the basis vector of the current pose of body and state of cloth, termed the temporal adaptive bases. For creating the low-dimensional bases of the pose space, (5) employed insight from a simulation train data to learn the pose space. However, this approach of learning from simulated data limits the actual dynamics of body pose and cloth state in cloth simulation. For instance, cloth wrinkles follow a semi-rigid shift in the directions of their folds and might be difficult to model using the principal component analysis. Hence, other bases vectors might suffice. Salazar et al. (2010) work focused on quantitatively analyzing the applications of cloth simulations with the multi-processor CUDA on GPU architecture. They further studied the number of iterations a collision treatment requires for axis-aligned bounding box hierarchies (9). Chenfanfu Jiang (2017) proposed a unique combination known as the Lagrangian/Eulerian approach that manages to select and retain the most essential aspects of these views (10). Their approach is for a real-time performance, and for that they considered only efficiency. The simulation software used is the PhysX and Bullet (11).

On the simulation software, most solutions provided by the graphics development companies are mostly plugin created on 3D packages like Maya, Unity or 3DS Max and solutions further exported as assets in formats readable by the game engine. An existing cloth simulator known as Shroud, with cutting-edge technologies, has been developed by CloakWorks. The powerful features of Shroud can be maximized via plugin to a game's engines and can simplify the creation of high-performance and overly realistic cloth in Unity-powered games. Also, methods such as hybrid, data driven, GPU implementation, position based dynamics and light weight CPU methods that in literature have been used to achieve real-time performance in cloth simulation (12). However, these methods are tailored towards either accuracy, efficiency or stability. This constitutes a challenge in the achievement of realism and the much-needed efficiency for real-time cloth simulation. It might not be a challenge in offline cloth simulations because only mechanical accuracy and realism is important. On the other hand, the real-time simulations require visual realism which is dependent on efficiency. In literature, PBD has been known to overcome the limitations of other methods. Also, it was used for PhysX real-time cloth (11). And it is recently used in Bullet engine. Therefore, this paper explores PBD. In subsequent section, the basic concept of position-based dynamics for cloth simulation will be discussed in detail.

### Proposed Method

Though much research effort has been made in cloth simulation, dynamically preserving cloth overlays and wrinkles is still a challenging task. This section explains the technical aspects of research. Since it is aimed at real-time performance, the algorithms are chosen to be as efficient as possible. This method was proven by the MAGIC-X center at University of Technology Malaysia. Constant density can be enforced by implementing a system where non-linear constraints are solved with one constraint per-particle. Each constraint consists of the particle's position and the positions of adjacent particles, which collectively is represented as  $\{p_1, \dots, p_n\}$ .

As explained by (6), every dynamic object in the simulation is represented by a set of N vertices and M constraints. The vertex  $i$  in  $[1, \dots, N]$  has a mass  $M_i$ , position  $P_i$  and velocity  $V_i$ . A constraint  $C_j$  consists of following:

A constraint  $j$  in  $[1, \dots, M]$  consist of

- A cardinality  $n_j$

- A function  $C_j: R_{3n_j} \rightarrow R$ ,
- A set of indices  $\{i_1, \dots, i_n\}$ ,  $i_k$  in  $[1, \dots, N]$
- A stiffness parameter  $k_j$  in  $[0, \dots, 1]$  and
- A type of either *equality* or *inequality*

where,

$n_j$  Cardinality defines the number of particles involved in the constraint,

$C_j$  Constraint function defines the relationship between the particles involved in the constraint.

$\{i_1 \dots i_n\}$  is the list of indices corresponding to the particles involved in the constraint.

$K_j$  represents the strength of constraint ranging from 0 to 1.

The constraint  $j$  with type equality is satisfied if  $C_j(p_1, \dots, p_n) = 0$  and type inequality is satisfied if  $C_j(p_1, \dots, p_n) = 0 \geq 0$ .

The steps of simulated dynamic object algorithm are defined below.

---

#### Algorithm 1: The steps of simulated dynamic object

---

```

1: for all indices  $I$ 
2: Initialize  $x_i = x_i^0, p_i = p_i^0, w_i = 1/m_i$ 
3: for all indices  $i$  do  $p_i \leftarrow p_i + \Delta t w_i f_{ext}(x_i)$ 
4: damp Indices( $p_{i1}, \dots, p_{in}$ )
5: for all indices  $i$  do  $p_i \leftarrow p_i + \Delta t w$ 
6: for all indices  $i$  do generate Collision Constraints ( $x_i \rightarrow p_i$ )
7: loop solver Iteration times
8: ProjectConstraints ( $C_1 \rightarrow C_{M+M_{call}}, P_1, \dots, P_N$ )
9: end loop
10: for all indices  $i$ 
11:  $p_i \leftarrow \frac{p_i - x_i}{\Delta t}$ 
12:  $x_i \leftarrow p_i$ 
13: end for
14: Velocity Update( $p_1, \dots, p_n$ )
15: end loop

```

---

#### Raised Cosine Function (RCF)

The raised cosine function (RCF) (13) is adopted for determining the list of constraints  $C_M + C_{call}$  and estimation of positions  $\{p_1, \dots, p_n\}$  as in (5). By using the RCF, the estimate positions can be modified in such a way that all the constraints of a given cloth fabric, both in a static and dynamic position, is satisfied. The resulting system of equation is non-linear. The influence of RCF on  $\{p_1, \dots, p_n\}$  range and each vertex, is as expressed:

$$F(\vec{p}) = \left(1 + \cos\left(\pi \times \frac{d_p}{d_{max}}\right)\right) \times w \times \Delta p \quad (1)$$

where  $p$  is the parameters towards a direction, 1297129712971297 is maximum displacement between points in the influence region,  $w$  is the weight of parameters on a specific region,  $\Delta p$  is the displacement of the control points provided by the value of parameter  $j$ , which is a set of parameters to be incremented for virtual human animation.

To deform parameters at the region of influence (1) is applied as the controlled coordinates are moved. This implies that the order of implementation is important because the effect is instantly visible, that is, the modified position at each step of the constraint projection is required to be available for the ensuing constraint projection process.

### Constraints Projection

Constraints are used to determine new velocities and positions of each vertex of a cloth. It takes stretching, bending, or external forces such as gravity, and collision avoidance, to generate constraints. The cloth simulation algorithm realizes new velocities and positions by point projection using constraint function of any of these constraint events. As much as interested in the projections, the angular and linear momentum should also be conserved. The constraint is derived as follows.

$$\sum p_i \Delta p_i = 0 \quad (2)$$

Where  $\Delta p_i$  is the displacement of indices  $i$  by the projection to satisfy the constraints. This results in a centered mass. To ensure that the angular momentum is conserved, the constraint is modified as follows.

$$\sum r_i \times p_i \Delta p_i = 0 \quad (3)$$

Where  $r_i$  is the distance of  $p_i$  to an arbitrary common influence region and all indices are treated to have equal mass.

The research estimates density estimation using Poly6 kernel and further show how density constraint can be incorporated into in most of the CPU powered games. Then the PBD is used to find a particle position correction  $\Delta p$  which satisfies the constraint. Since projection is along the direction of maximal change, the direction will be perpendicular to rigid body modes. Also, the gradient,  $\nabla p C$ , is in similar direction. Here, linear and angular moments are automatically conserved if the correction  $\Delta p$  is chosen to be along  $\nabla p C$  and given equal masses. Therefore, given  $p$ , the correction  $\Delta p$  is chosen such that  $C(p + \Delta p) = 0$ . This is estimated for each vertex point,  $p_i$  as follows:

$$\Delta p_i = -s \nabla p_i C(p_1, \dots, p_n) \quad (4)$$

The scaling factor,  $s$ , is the same for all points. Moreover, the corrections  $\Delta p_i$  can be weighed by the inverse mass  $w_i = 1/m_i$  if each of the point is of different mass. However, if a point is fixed, the mass of the point is simply set to infinity. With an inverse mass of 0, it translates to corrections  $\Delta p_i = 0$ , and the estimate position remaining the same, that is, the point is at origin. Therefore, Equation (4) can be replaced by:

$$\Delta p_i = \lambda w_i \nabla p_i C(p) \quad (5)$$

Finally, the cloth model on human avatar, as shown in Fig. 1, is formed from:

$$\Delta p_i = -s_x w_i \nabla p_i C(p_1, \dots, p_n) \quad (6)$$



Figure 1. Real time simulated cloth with constraints projection

### Damping

To preserve the rigid body modes of the object, global damping is applied. The algorithm is presented as a flowchart in Fig. 2.

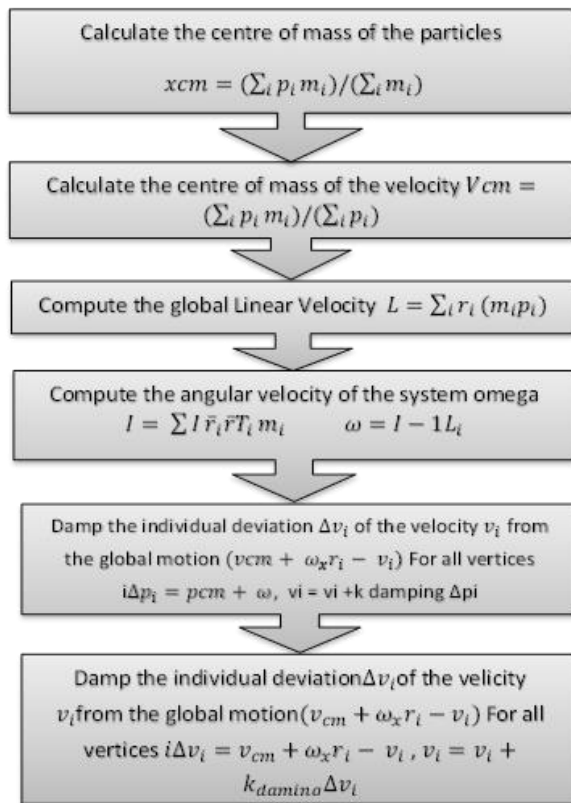


Figure 2. Global damping that added to the object

For clarity  $r_i = x_i - x_{cm}$ , and  $\bar{r}_i$  is a  $3 \times 3$  matrix with the property  $\bar{r}_i v = r_i v$ , and  $K_{damping}$  is bounded by  $[0, 1]$  which is the damping coefficient. The objective is to preserve the rigid body modes of the object because in the extreme case where  $K_{damping} = 1$ , the set of vertices behave like a rigid body, thus, only the global motion endures. For arbitrary values of  $K_{damping}$ , the velocities are globally dampened without affecting the global motion of the vertices.

### Cloth Representation

Algorithm 1 provides an outline to the simulation loop. The research presents the PBD with the exception that each constraint is solved independently in a raised cosine deformation function (RCDF). As part of the constraint solving loop, collision constraint detection will be performed.

As earlier identified, stretch and bending constraints are two types of static constraints that can reproduce natural cloth behavior. These constraints represent the stretchy and bendy nature which characterizes a cloth. The stretching constraint is added as a distance constraint function because distance constraints preserve the stretching behavior of cloth. For the bending constraint, there are two methods; dihedral bending constraint Muller et al (2007) and other one is called Triangle bending constraint as

proposed by (11). These methods are discussed subsequently.

### Stretch Constraint

The stretch constraint is similar to a spring, it maintains the particles within a minimum distance that connects it. Thus, it addresses the stretch-stiffness of cloth. By implication, the constraint has to be projected throughout the simulation unconditionally, thereby resulting in equality constraint. The constraint function is as expressed as follows and its property demonstrated in Fig. 3.

$$C(p_1, p_2) = |p_1 - p_2| - d \quad (7)$$

With stiffness  $K_{stretch}$  and  $d = rest\ length$ . Using the projection constraint function, deriving the gradients of the function with respect to  $p_1$ , and  $p_2$ , derive:

$$\nabla p_1 C(p_1, p_2) = n \text{ and } \nabla p_2 C(p_1, p_2) = -n \quad (8)$$

where  $n = \frac{p_1 - p_2}{|p_1 - p_2|}$

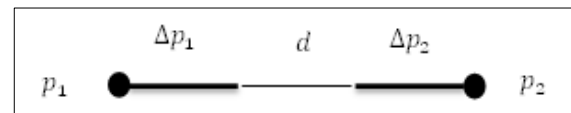


Figure 3. Projection of the constraint between  $p_1$  and  $p_2$  and minimum distance  $d$

Figure 4 below illustrates the behaviour of the PBD cloth model for various stiffness values for the two constraints. In case (a) the cloth is stretchy for a stretching stiffness value of 0.5 in case (b) the cloth looks good with high stiffness stretching constraint and low stiffness bending constraint.

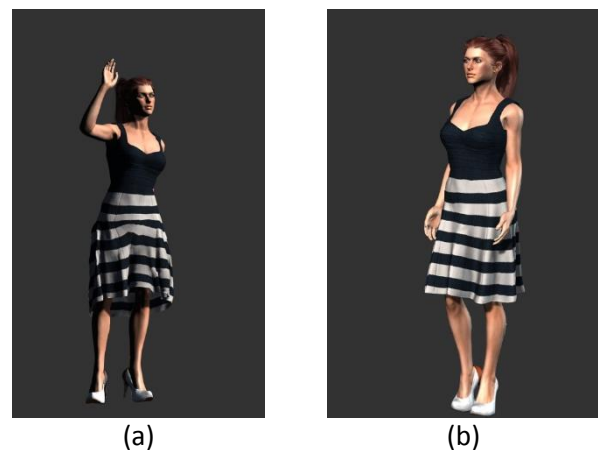


Figure 4. Comparison of real time cloth behaviour based on the constraint stiffness

### Bending Constraint

The bending model which was proposed by Kelager et al. (2010), began with the idea of collapsing a triangle using as few and cheap operations as possible (11). By this, potential energy can be introduced where possible and uses the centroid of a triangle (see illustration in Fig. 5) for defining a constraint. The three medians meet exactly at the centroid in a ratio 2:1. In order to collapse the triangle while conserving the linear and angular momenta, the base line vertices  $b_0$  and  $b_1$  are displaced by  $\frac{1}{2}(v-c)$  and whereas  $v$  is updated as  $v = c$ . The distance between the edge of the triangle  $v$  and the centroid  $c$  is used to generate the constraint function as follows:

$C_{triangle}(b_0, b_1, v) \geq \|v - c\| - (k + h_0)$  (9)  
where  $k$  is a global bending parameter,  $h_0$  is the length of rest, and  $c = \frac{1}{3}(b_0 + b_1 + v)$ , which is the center of medians of the triangle, is the inequality projection constraint whose projection is evaluated only when  $C_{triangle} < 0$ .

$$b_0 = \frac{2wb_0}{w}(v - c)\left(1 - \frac{h_0}{\|v - c\|}\right),$$

$$b_1 = \frac{2wb_1}{w}(v - c)\left(1 - \frac{h_0}{\|v - c\|}\right), \quad \text{and}$$

$$v = \frac{4wv}{w}(v - c)\left(1 - \frac{h_0}{\|v - c\|}\right).$$

The  $w$  is the inverse mass for triangle,  $w$ , where  $w = w_{b_0} + w_{b_1} + 2w_v$ .

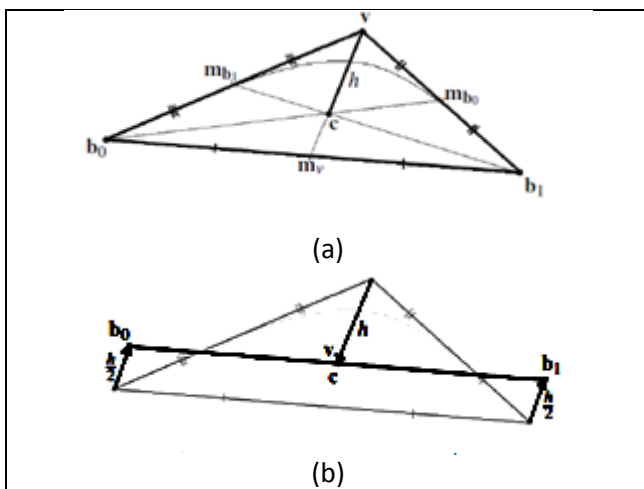


Figure 5. The triangle bending model (a) The intersection point of the three medians is the centroid  $c$ . The radius of curvature is represented by the length  $h$  from the centroid to the tip of the triangle. By employing the length difference, a simple geometric constraint can be formulated. (b) The triangle in a collapsed state where the vertices have been displaced along the direction of the center median (11).

### Collision Constraints

Unlike the stretch and bending constraints, collisions are dynamic constraints created in PBD which changes at every time step. In cloth simulation, collision is of three perspectives; plane collision, sphere-capsule collision and self-collision. The plane collision detection is used to check the collision of the object with the floor to keep the cloth from falling infinitely. Once a contact collision is established the point may be moving in a velocity towards the surface or away from the surface. Thus, a further check is done to determine if the velocity is away from the surface (11).

The sphere-capsule collision uses collision volume to create a game character. This type of collision which generates bounding capsules on game characters' body parts is mostly appropriate for game characters and is commonly found in most of the CPU powered games. It is important to note that the sphere-capsule collision performs detection based on a negligibly radius assigned to the sphere that represents the cloth particles. The self-collision also termed point triangle collision requires spatial partitioning. It detects particles and triangles of itself or other cloth meshes. Therefore, wind usually acts on faces instead of individual particle, but uses the surface normal at each point to modify face. The effect of self-collision is as illustrated in Fig. 6 and the force which acts on the face of the triangle is expressed as:

$$F = c * (n, d) * n \quad (10)$$

where  $c$  = wind strength,  $d$  = wind direction  $n$  = the normal of the face at a point.



Figure 6. The direction of forces wind (a) Apex PhysX cloth simulation by NVIDIA Developer (b) The direction of forces acting on a triangle face due to wind

### Experimental Results

For the experiment, the APEX clothing, APEX C and NCCA Graphics Library. The APEX clothing (PhysX Clothing) library is used which is primarily for cloth authoring in real-time simulation of dynamic clothing in the development of game character. The APEX C is part of an NVIDIA APEX framework, while NCCA Graphics Library

is an open-source library for actualizing shading, and acts as a storage container for graphical data such as vectors, matrices, and faces.

The research used static and animated character meshes to cloth the human avatar by considering the physics parameters which are manipulated based on the asset imported for desired outputs. Further, the real-time performance is analyzed by considering the number of particles in the system, the quantity of cloth constraints and the presence of self-collision. The human avatar is an animated mesh modelled to wear a cloth

constructed using 1218 particles without self-collision and dynamically exhibit motion through dance animation. The reason is to reflect different characteristics of cloth on motion. The research utilizes a frame rate of 6 FPS, which is twice the original FPS to achieve unaltered cloth response and the performance runs at a speed above 40 FPS. To estimate neighbors, MPEG4 as adopted in Ali et al. (2018) is used (5). The resulting output of the process on the animated mesh is illustrated in Fig. 7.

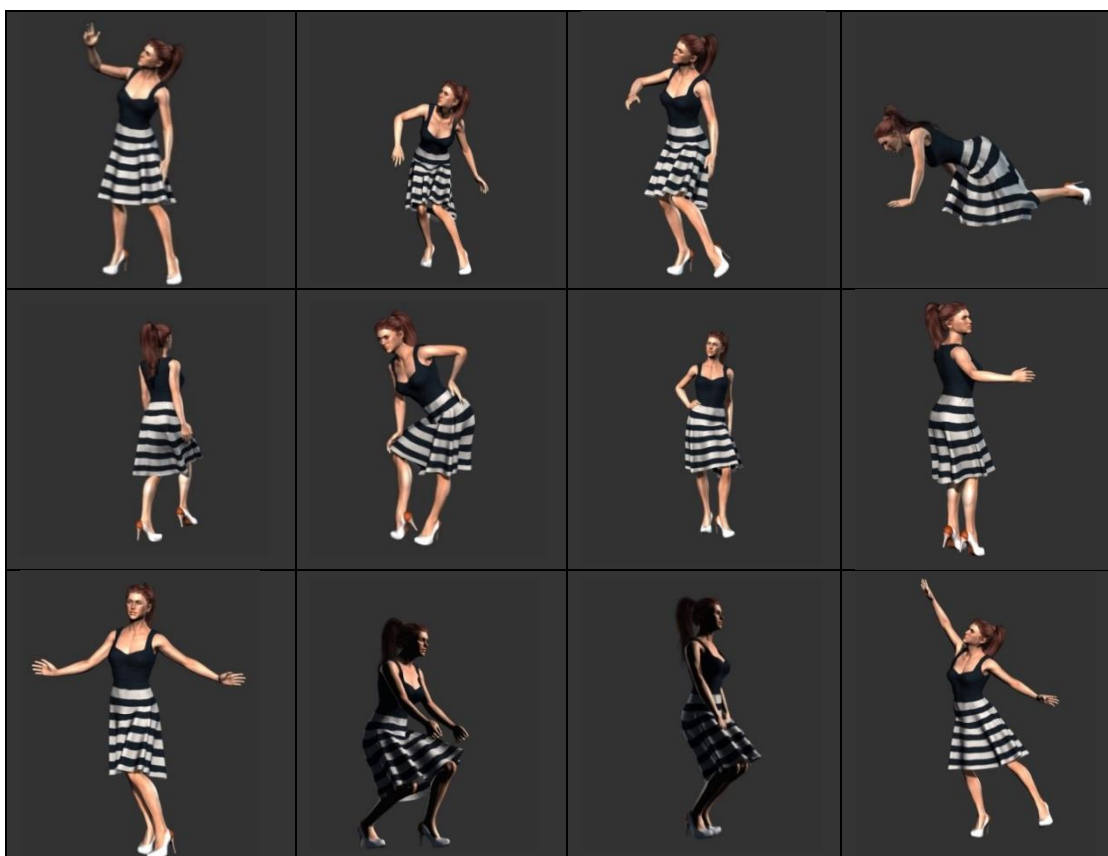
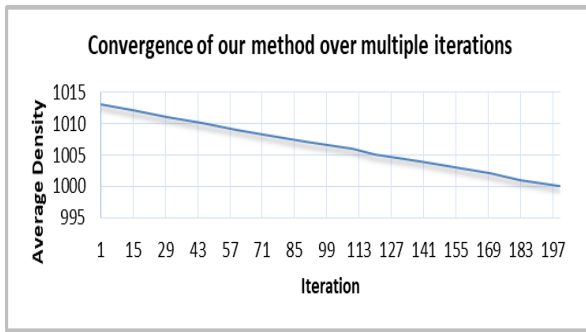


Figure 7. Results achieved through the proposed method

Cloth simulation with self-collision turned on. The system has 1030 particles and 6470 cloth constraints which simulated at 20 FPS. Since dance animation is used to reflect the nature of cloth in the presence of wind, the research simulated wind effect on the cloth. For continuity of comparison of results in this area, the research simulated wind effect using our proposed method and compare it with results of wind simulation using Apex PhysX cloth simulation by NVIDIA. As can be observed in Fig. 8, Apex PhysX is not stable at less than 10 sub-steps per frame ( $\Delta t = 0.0016$  s) whereas with our proposed method stability is achieved in a single

timestep ( $\Delta t = 0.020$  s). Also, to show a comparison with respect to compressibility, the research runs Apex PhysX with 10 sub-steps and 4 iterations and with ours the research runs 4 sub-steps and 10 iterations per sub-step. In this way, each performs a total of 50 iterations per frame. Our motive is to demonstrate that our proposed method can accomplish similar results as with Apex PhysX at higher time-steps. In essence the research is able to demonstrate the per step expenses of grid constructions and neighbour findings over more density iterations.

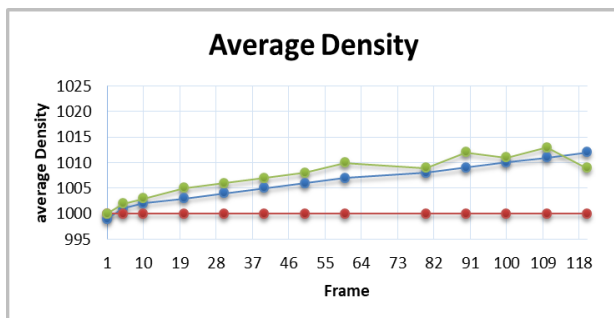


**Figure 8. Convergence of our method over multiple iterations**

Further, the research presents in Fig. 9 the convergence of our proposed method over multiple iterations. The graph which is a plot of density against the number of frames in the simulation suggests a promising outcome of the proposed method. The results demonstrate that the level of compression of our proposed method is comparable to Apex PhysX, though ours uses a larger time step (see in Fig. 10). However, to circumvent this shortcoming, each phase of the algorithm can be made completely parallel though the use of parallel architectures like the GPUs.



**Figure 9. Results achieved through forward and backward movements.**



**Figure 10. Density during the bunny drop simulation. Our algorithm retains compressibility comparable to PCISPH at time steps twice higher. Colour key: red stands for rest density, Green stands for Apex PhysX and Blue represents our method.**

The use of a PBD based approach shows to be beneficial for the simulation of deformable cloth entity and is shown to make simulation run at faster rate. Now on plugin, the options to set the physics property of the cloth are provided within the Unity plugin and aids in the preview of the simulation to ease the authoring process. However, a few shortcomings are identified: 1) when the scene is loaded back, the settings should also be loaded back to enable resume of work to most current state, 2) texture maps can be used to represent attachment points on cloth mesh rather than importing their position data, because it can become inefficient when working with larger assets, and 3) the cloth is

currently attached to the moving capsule but should instead be attached to the animated mesh.

The image on the left is Havok Previewer application that previews the assets imported from Havok Cloth authoring tool. The image on right is the developed application that simulates and visualizes the assets imported from the Maya cloth Authoring tools. As illustrated in Fig. 11, the image (a) is Havok previewer application that previews the assets imported from Havok Cloth authoring tool. The image on (b) is the developed application simulated image.



(a)



(b)

**Figure 11. (a) Havok asset viewer and (b) the developed real time simulator Application with cloth resolution: 20X40 at 20 iterations**



## Conclusion:

The research developed a real-time cloth simulator to model clothing on a virtual human character using a PBD based approach. The PBD is enhanced by computing a series of positional constraints which implemented constant densities. The proposed method enables comparable incompressibility and convergence to RCD solvers. The result achieved is a real-time cloth simulation with physically plausible outputs on animated virtual human character with worn cloth at different wind effect. Hoping that our method serves as a completion to the game assets clothing pipeline. The main contributions of the paper are: is the real-time performance of the simulator with optimized collision in clothes, the syncing of the animation with the cloth simulation and manipulating the properties of the cloth for best output possible. Finally, the self-collision and collision with moving capsules achieved realistic 3D cloth model and cloth behavior on animated characters in real-time. However, our method is limited by the inability to model solid objects with cloth due to lack of the inclusion of spring-constraints in the process.

## Authors' declaration:

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are mine ours. Besides, the Figures and images, which are not mine ours, have been given the permission for re-publication attached with the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee in University of Baghdad.

## References:

1. Fabian H, Bernhard Th, Stelian C, Robert WS, Forrester C, Mark M, et al. Subspace Clothing Simulation Using Adaptive Base. *ACM TOG*. 2014; 33(4): 1-9.
2. Shi Y, Long P, Xu K, Huang H, Xiong Y. Data-driven contextual modeling for 3D scene understanding. 2<sup>nd</sup> ed. *Computer and Graphics*, Elsevier. 2016; 55: 55–67.
3. Bender J, Deul C. Adaptive cloth simulation using corotational finite elements. *Comput Graph*. 2013; 37: 820–829.
4. Lu J, Zheng C. Dynamic cloth simulation by iso geometric analysis. *Comput Methods Appl Mech Eng*. 2014; 268: 475–493.
5. Ali I R, Kolivand H, Alkawaz M H. Lip syncing method for realistic expressive 3D face model. *MULTIMED TOOLS APPL*.2018; 77(5): 5323-5366.
6. Muller M, Heidelberger B, Hennix M, Ratcliff J. Position Based Dynamics. *JVCIR*. 2007; 18(2): 109-118.
7. Long J, Burns K, Yang J. Cloth Modeling and Simulation: A Literature Survey. *Lecture Notes in Computer Science* [internet]. 2011: 312–320. Available from: 10.1007/978-3-642-21799-9\_35
8. Vassilev TI. Comparison of several parallel API for cloth modelling on modern GPUs. *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies*, ACM, University of Ruse, Bulgaria. 2010; 131-136.
9. Salazar FSR, Machado BB, Oca A. De Oliveira MCF. Cloth simulation using AABB hierarchies and GPU parallelism. *Games and Digital Entertainment (SBGAMES)*, IEEE Xplore, Rio de Janeiro, Brazil. 2010; 97–107.
10. Chenfanfu J, Theodore G, Joseph T. Anisotropic Elastoplasticity for Cloth. *Knit and Hair Frictional Contact*. *ACM Transactions on Graphics (SIGGRAPH 2017)*. 2017; 36(4): 1-152.
11. Kelager M, Niebe S, Erleben K. A Triangle Bending Constraint Model for Position-Based Dynamics. *VRIPHYS*. 2010;10:31-7.
12. Capitán-Vallvey, López-Ruiz N, Martínez-Olmos A, Erenas MM, Palma AJ. Recent developments in computer vision-based analytical chemistry: A tutorial review, *Analytica Chimica Acta*. 2015; 899: 23–56.
13. Leprince S, Barbot S, Ayoub F, Avouac JP. Automatic and precise orthorectification, coregistration, and subpixel correlation of satellite images, application to ground deformation measurements. *IEEE Trans. Geosci. Remote Sens*. 2007; 45: 1529–1558.

## محاكاة القماش في الوقت الحقيقي على الشخصية البشرية الافتراضية باستخدام تقنية الإطار الديناميكي المعزز القائم على الموقع

هوشانك كوليفاند<sup>4</sup>

طالب الشريفي<sup>3</sup>

مازن عبد محمد<sup>2</sup>

اعتماد رحيم علي<sup>1</sup>

<sup>1</sup>قسم ادارة انظمة المعلوماتية، كلية معلوماتية الاعمال، جامعة تكنولوجيا المعلومات والاتصالات، بغداد، العراق.

<sup>2</sup>قسم علوم الحاسبات، كلية علوم الحاسبات وتقنيات المعلومات، جامعة الانبار، انبار، العراق.

<sup>3</sup>قسم هندسة الاتصالات، كلية الهندسة، كلية الرافدين الجامعة، بغداد، العراق.

<sup>4</sup>قسم علوم الحاسبات، كلية الهندسة والتكنولوجيا، جامعة ليفربول جون موريس، ليفربول، المملكة المتحدة.

### الخلاصة:

كانت محاكاة القماش والرسوم المتحركة موضوعاً للبحث منذ منتصف الثمانينيات في مجال رسومات الكمبيوتر. إن فرض عدم الضغط مهم جداً في محاكاة الوقت الفعلي. على الرغم من أن هناك إنجازات كبيرة في هذا الصدد، إلا أنها لا تزال تعاني من استهلاك الوقت غير الضروري في خطوات معينة شائعة في التطبيقات في الوقت الفعلي. يطور هذا البحث محاكاة قماش في الوقت الفعلي لشخصية بشرية افتراضية مرندية ملابس. وقد حققت هذه المخطوطة الاهداف في محاكاة القماش على الشخصية الافتراضية من خلال تعزيز إطار الديناميكيات القائمة على الموقع من خلال حساب سلسلة من القيود الموضعية التي تنفذ كثافات ثابتة. أيضا، يتم تنفيذ التصادم الذاتي والاصطدام مع الكبسولات المتحركة لتحقيق قماش سلوك واقعي على غرار الرسوم المتحركة. وذلك لتمكين عدم قابلية المقارنة والالتقاء مقارنة بمذبيبات دالة تشوه جيب التمام عند التنفيذ، نحقق تصادمًا محسنًا بين الملابس، ومزامنة الرسوم المتحركة مع محاكاة القماش وتحديد خصائص القماش للحصول على أفضل النتائج الممكنة. لذلك، تم تحقيق محاكاة القماش في الوقت الحقيقي، مع إخراج معقول، على الشخصية الافتراضية المتحركة. ندرك أن طريقتنا المقترحة يمكن أن تكون بمثابة استكمال للبحوث السابقة في حقل ملابس الشخصيات الافتراضية.

**الكلمات المفتاحية:** محاكاة الملابس، قيود التصادم، قوة الحركة، ديناميكيات القائمة على الموقع، دالة التشوه جيب تمام.