Collaborative and Human Based Performance Analysis

Mathew Schwartz
Advanced Institutes of Convergence Technology, South Korea
www.smart-art.org
mat@cadop.info

Abstract. This research presents methods for simulation and visualization of human factors. This allows for a performance based analysis of buildings from the local human scale to the larger building scale. Technical issues such as computational time and mathematically describing a buildings geometry are discussed. The algorithms presented are integrated in a 3D modeling software commonly used in design and architecture through a plugin.

Keywords. Universal design; human analysis; collaboration; education; disability.

INTRODUCTION

In general, the field of Building Information Modeling (BIM) is related to structural or environmental analysis. This is likely due to the main users of BIM: Architecture, Engineering, and Construction (AEC). By utilizing BIM approaches these fields are able to reduce human error and cost from design to construction (Azhar et al., 2008). While BIM can include many forms of analysis, human factors have not traditionally been included. Additionally, the definition of BIM is still being debated and developed (Eastman, 2008). For this reason, this paper concentrates on the term building performance to discuss the relationship of human factors to overall building analysis. However, this term is only for clarity, and the work presented is well suited to be integrated as a feature of BIM software.

As with BIM, building performance is largely focused on structural and environmental factors that can be quantified. When the human is included in an analysis of building performance it is usually related to the environmental effects of a human, or multiple humans, such as heat transfer or acoustical properties (Mahdavi, 2011). There are a few reasons the ergonomic and physiological aspect of humans are not usually considered necessary to include. For one, many important aspects of a building in regards to humans are regulated in laws, for example, the American Disabilities Act (ADA) (Dept. of Justice, 2010). Similarly, many of the components related to humans are standardized, which has been needed as the AEC community is not focused on researching ergonomics or biomechanics, large research fields in themselves. Additionally, excluding emergency situations, the cost of an error in construction is much greater than a door knob being difficult to use. While these reasons are valid, they only represent the minimum of what design can be.

The philosophy of Universal Design brings to question the lack of human factor analysis for building performance. Exclusion of human factors from the design and analysis stage while relying on standards and prescribed law can be stigmatizing and may need to be fixed later, demonstrating the failure of design (Story, 1998). While the ADA is specific to disabilities, Universal Design is meant to benefit all people. Universal Design has become part of many curriculums in architecture schools (Vance, 2012), however, understanding the problems with designs
related to human factors can require costly physical experiments or communication with experts in other fields. Schubert et al. (2011) present a method for integrating physical tools with the design workflow, and Maver et al. (2001) demonstrate the ways in which universal design concepts can be explored through virtual reality. Although both of these present alternative methods to real world experimentation, they still require physical space and money.

This paper is meant to function as a reference on the types of quantifiable human factors, an approach to education, and a method for collaboration between the complicated fields of ergonomics and biomechanics with architecture and design. A 3D Manikin plugin is presented with a graphical user interface allowing access to the underlying algorithms that simulate and visualize human factors of the manikin with the space at different scales. Algorithms written in python are presented for five main areas: Reach, Vision, Zone, Search, and Movement. Aspects of biomechanics software are integrated with the plugin to create a platform for collaboration. To cohesively demonstrate the application of each component a manikin in a wheelchair is used.

**METHODOLOGY OF COMPONENTS**

There are many components to building design that effect human performance. The methods described in this section describe both human factors that should be addressed as well as methods in which to perform the simulation, however, this list is not exhaustive. The methods describe the algorithms used in the plugin written for MAYA, but can be applied in almost any 3D modeling tool.

**Reach**

While every person is slightly different in their ability to reach, the effects of a disease or spinal cord injury on reach ability (Jacquier-Bret et al., 2008) is important to know. A large problem with the ADA standards is the assumption that a person in a wheelchair is only limited in leg functionality. As this is not true, the ability to simulate reach ability for different users allows for a performance based analysis of the environment by comparing the reach ability to desired reach locations. The inverse kinematic (IK) system used to control the manikin arm was developed in MAYA, however, additional research into disability reach would greatly improve the simulations. Currently one of the most used and advanced simulation tools for ergonomics uses a reach envelope, visualizing the extent of a persons reach (Blanchonette, 2010). This is generally okay when a user is able to rotate to one side or another, however, if the user has a specific type of reach ability a more complete map needs to be simulated.

A voxel volume was used to create a complete 3D map of the reach ability. Each voxel is given a value based on the ability for the IK solver to find a solution for the voxels around the initial voxel as seen in equation (1).

\[
\vec{r} = n_x \hat{x} + n_y \hat{y} + n_z \hat{z}, \text{ where } \vec{r} \in V
\]

\[V \text{ is the set of all points in the volume.}\]

\[
\vec{r}_i = l\hat{x} + m\hat{y} + n\hat{z}, \text{ where } l, m, n \in \{-1, 0, 1\}
\]

\[
f(\vec{r}) = \sum_{\{\vec{r}+\vec{r}_i|\vec{r}_i \in N(\vec{r})\}} 1
\]

Where \(N(\vec{r}) = \{\text{reachable neighbors}\}\) (1)

Physically, this means if a persons hand is in an initial position, that position is ranked based on how many ways the hand can leave that position. The summation over 1 can be modified for a more elaborate or specific analysis of the reach ability volume. For instance, by changing 1 to the distance from the shoulder multiplied by gravity, areas further away from the body will be given lower scores than those closer. In the case of spinal cord injuries, this value can reflect the joint angles required to reach a position. The visualization corresponds to the simulation results such that opaque green voxels are the highest ranking and transparent red voxels are the lowest ranking. The method for coloring voxels, without transparency, is similar to robotics work in which the workspace must be defined and visualized (Zacharias et al., 2007). The visualization can be seen in the results section.
Vision
Designing with a persons vision in mind requires two aspects, items that should be seen, and items that should not. The former case includes items such as signs and navigational cues, while the latter includes direct lighting. An important aspect of vision is neck mobility. If for instance a person is in a wheelchair, the range of movement in the neck may be limited. In this case, as well as in many elderly people, it cannot be assumed that a building occupant will turn their head to view an item. In the case of direct lighting, many times a person will be uncomfortable, or worse, unable to see in the distance resulting in a dangerous situation.

The importance of vision is well known in architecture and design. In Tilley et al. (2001), a diagram showing many aspects of the human vision is used as a reference for designers. The problem with the diagram is the difficulty in understanding and translating the content to ones own design within both practice and education. The diagram can be represented in 3D space by referencing both top and side maximum angles. For displaying the vision regions each one of the four points, referred to as Limit, that describe the region are input at both an initial distance and the distance to where the visualization should end. Equation (2) shows the formula to calculate where each point should be placed.

$$\sqrt{\left(\frac{\text{distance}}{\sin(90 - \text{Limit})}\right)^2 - \text{distance}^2}$$  \hspace{1cm} (2)

In order to analyze the building for direct lighting at a specific point the location of the manikin head as well as direction is calculated in regards to each light. According to the diagram in Tilley, the disability glare zone is 45 degrees above the persons eye level and the extent a person can see to the side is 94 degrees. The domain is found with two calculations. First, the angle above the eye level is found creating a right triangle between the head position and light position with the hypotenuse being the distance between the two points. The adjacent side is the horizontal distance and the opposite side is the vertical distance. The algorithm for determining if the light is within the vertical 45 degree angle is seen in (3).

$$\tan^{-1}\left(\frac{\text{Horizontal Distance}}{\text{Vertical Distance}}\right) < 45$$  \hspace{1cm} (3)

Second, the location of the light relative to the head must be found. For this, a reference point is placed at any distance in front of the head in the direction it is facing. A triangle is constructed from the head position, reference position, and light position with the light projected onto the head up axis, assuming y is the up axis (4).

$$a = \sqrt{(H_X - R_X)^2 + (H_Y - R_Y)^2 + (H_Z - R_Z)^2}$$
$$b = \sqrt{(R_X - L_X)^2 + (R_Y - H_Y)^2 + (R_Z - L_Z)^2}$$
$$c = \sqrt{(L_X - H_X)^2 + (L_Z - H_Z)^2}$$  \hspace{1cm} (4)

where $H$ is the Head position, $R$ is the reference position, and $L$ is the light position. Using the law of cosines, the side angle is calculated and checked against the 94 degree limit (5).

$$\cos^{-1}\left(\frac{c^2 + a^2 - b^2}{2ca}\right) < 94$$  \hspace{1cm} (5)

The lights found to be in the disabling glare zone can be marked in the model and the total number can be displayed for the designer.

Zone
The space around a person can be classified in multiple ways. There is the space someone needs to be physically present, space that makes someone comfortable, and space in order to complete a task. The latter two are most often related to psychology as well as physiology. The importance of the space around a person can be seen in design reference books (Zelnik and Panero, 1979) as well as in research papers (Lantrip, 1993). When designing the spatial needs of a wheelchair the designer must understand the many situations that exist. For example, if a hallway is designed to be wide enough for a wheelchair to pass through, it may not be wide enough for the wheelchair to turn around, or more commonly, for a person in the opposite direction to pass by.

The implementation of a spatial zone is relatively easy in a CAD program. The spatial zone needed can be visualized by a transparent cylinder around
the manikin as seen in the results section. In the case of multiple people more than one cylinder can be displayed. The coordinates in the mesh are generated the same as in (2) where Limit is defined as the distances from the center of the manikin. As the generated mesh contains a Cartesian domain, it is possible to analyze environmental interference in regards to the generated geometry.

**Search**

Out of many human factors that can be included in building analysis, the distance and way in which someone moves throughout is one of the most difficult to integrate, yet is extremely important. While laws exist for wheelchair ramps and elevators, knowing how a building layout or size affects the occupants is required to design above the minimum. There are many forms of analysis of human factors that are useful at the human scale, including egress and accessibility. Two components needed to analyze movement throughout a building are: mathematically describing the building and creating the path of movement. From this it is possible to run numerous simulations of human factors to analyze the path created. The current implementation refers to the path as the energy required.

The easiest way to describe a building using a node graph is to make landmark areas nodes and create connections, known as edges, between them. When the fastest way between two nodes is desired, a search algorithm calculates the edge values from one point to the other (Dijkstra, 1976). In general, the distance between each landmark is the value given to the connection. However, the distance between each landmark must be the length of the route from start to end, not the Cartesian distance. This creates a problem when attempting to translate a 3D model of a building into a searchable graph without individually measuring each path (Figure 1).

Using the internal raytracing function of MAYA, a technique was developed for mathematically describing a 3D model of a building. The first assumption is that any geometry that should be considered as ground should begin with the naming convention “floor”. Anything following that name is irrelevant and can be used for the designers own organization. The second assumption is that the manikin is placed over any valid ground. If these conditions are satisfied, the algorithm will send a ray from the top of the manikin head to the floor and return the Cartesian position. Each valid ray is checked against the minimum required space, in this case, a wheelchair turning radius. The algorithm then expands outward and continues to store valid node locations as seen in (6).

Let \( C_i = \{ \vec{r}_k \mid |\vec{r}_k - \vec{r}_i| = R \} \)

Where \( R \) is the wheelchair turning radius.

\[ N = \{ \vec{r}_i \mid \forall \vec{r}_k \in C_i, \vec{r}_k \text{ is on the floor} \} \]

\( N \) is the set of valid nodes. Let \( \vec{r}_{ij} = \vec{r}_j - \vec{r}_i \) and \( a \) be the spacing between nodes in the \( x, y \) plane.

The edge values are the key to analyzing the building. As each node is created, so are the edge costs to each of the available nodes surrounding it (7).

\[ E = \{ (\vec{r}_i, \vec{r}_j) \in N \times N \mid (\vec{r}_{ij} \cdot \hat{x})^2 + (\vec{r}_{ij} \cdot \hat{y})^2 = a^2 \} \]

\( E \) is the set of edges.

\[ f(\vec{r}_i, \vec{r}_j) = g(\theta_{ij}) \text{ where} \]

\[ \tan \theta_{ij} = \frac{\vec{r}_{ij} \cdot \hat{z}}{\sqrt{(\vec{r}_{ij} \cdot \hat{x})^2 + (\vec{r}_{ij} \cdot \hat{y})^2}} \]

\( g(\theta_{ij}) \) is a value function that assigns energy cost to each edge based on the angle of inclination.
Altering the edge cost can be useful to analyze different situations such as the path of a wheelchair or a person with crutches. For example, implementing this algorithm for a wheelchair would be done by making any edge that is at a slope, larger than the maximum required by law, an infinite value. By returning a path through the nodes, the physical distance can also be calculated. As each node has a total value assigned to it by the connecting neighbors a visual representation can also be created.

Movement
As previously mentioned, one problem with fully integrating human factors in design workflows is the extra knowledge of human biomechanics required of designers. As both a teaching and collaboration tool, this research demonstrates a method for integrating the two fields.

Many types of analysis in biomechanics use joint angles. To integrate this workflow a python library for creating matlab style graphs (Hunter, 2007) was implemented within the plugin. As some design modeling tools do not have animation features, all time related elements are stored in comma separated value (CSV) files. The file structure implemented is the same as those from the exported joint angles of biomechanics software.

**Graphical User Interface**
Combining the algorithms together and linking them to the 3D modeling software is done through callbacks and signals within a graphical user interface (GUI) (Summerfield, 2007). The GUI is designed to give the designer access to all of the variables within the algorithms. Each component is given a separate tab in the GUI with a full range of settings and variables (Figure 2).

**RESULT**
The implementation of the algorithms resulted in a cohesive tool that integrates with a 3D model of a building. A variety of visualizations, simulations, and analysis are available for the designer to use. As the building is being designed the only requirement of the designer is to place the manikin in a desired location. Once the manikin is placed, the designer can use any of the default settings to run the simulations. As shown by Eriksson et al. (2000), there is great value in participatory planning. This plugin allows for participatory planning by creating a second window in which the view of the manikin is displayed (Figure 3).

As it is not realistic to expect all architecture schools to have time for real world experiments, or access to wheelchairs, it is important to have a virtual method of exploring the same topics. During the early stages of this research a Universal Design
class led by Sean Vance at the University of Michigan was documented (Vance, 2012). The students were not aware or directly influenced by this research. Multiple physical experiments were conducted by the students to simulate the relationship of disabilities to the built environment around them. While it is very useful for students to have a hands-on knowledge it may not always be possible. The tool presented here is capable of simulating and visualizing the same ways in which students felt the problems needed to be addressed. Some of these issues include obstacles in the way of a wheelchair (Figures 4 and 5), the line of sight from a person in a wheelchair (Figure 6), and items out of reach (Figures 7 and 8).

While it is possible to quantify all of the presented human factors relative to a building model, the largest scale quantifiable factor is in navigation. The graph search algorithm is very flexible and can be used for a variety of situations, one of which is seen

Figure 4
Left: Student points out objects in the way of navigation. Top-right: the path from manikin to a location is drawn. Bottom-right: objects placed over the ground are accounted for, creating a path that avoids the object.

Figure 5
Left: Student points out problems with the distance between toilet and wall for turning radius. Right: Wheelchair dimensions are visualized in a 3D model of a room.
Figure 6
Left: Student draws line of sight to point out the difficulty in communication due to the high partition. Right: Two different vision cones are visualized. Two perspectives are shown, one from the building participant and a normal 3D model view.

Figure 7
Left: Student shows problems with the placement of different items in a room. Right: The reach ability map is simulated and visualized with transparency and color.

Soap dispenser is too high and inaccessible
Mount soap dispenser here instead

When taller individual uses shower and does not replace wand in lowest position, wand becomes inaccessible
A pressure controlled wand might be an appropriate solution, where while the water pressure is on, the wand stays in place. Once water pressure is off, wand slides down to lowest position

in Figure 4. Another aspect of the graph search is the ability to analyze the amount of time it takes for someone to move from one point to another. In the case of a wheelchair, depending on the type of dis-
ability and wheelchair, a person will move between ~50 and ~80 meters per minute (Beekman et al., 1999). Taking data from a subject with paraplegia in a standard wheelchair a rate of 75 meters per minute can be used for analysis. When a designer runs the search algorithm the GUI will display the amount of time it takes to move along that path (Figure 9).

In addition to the speed of movement it is possible to calculate the energy expenditure of a movement. While there is a lack of human subject testing that can give an accurate simulation for all terrain, some basic estimations can be made. This approach can be used when designing a ramp and the designer is deciding if the ramp should be short, with the maximum allowed slope, or longer with a lower slope. As the search algorithm will find the lowest cost method to get to the end point, if the algorithm creates a higher cost of movement for a steeper incline, the path generated will reflect the best method. This situation is ideal for bringing in a biomechanics expert, or in the case of education, biomechanics students. As the biomechanics side is able to analyze the movement of a person during wheelchair propulsion and quantify the results, the design of the ground floor can be changed. Using this alongside the colored visualization shown in Figure 8 creates an opportunity for informed iterative design.

CONCLUSION
This paper presents a variety of functions, algorithms, and systems that can better integrate human factors with the design workflow. Each algorithm has potential to be both improved with speed and expanded on for functionality. The significance of the work is not limited to BIM and Building Analy-

Figure 8
Left: Student drawing angles of human body to analyze limits. Right: Motion capture data of reach is displayed through a graph.

Figure 9
Left: Student demonstrates the narrow hallway does not allow for a wheelchair to turn around. Student notes the length of the hallway with no outlets can cause arm fatigue. Right: GUI Displays estimated time to travel using a wheelchair. Color coded ground shows results of simulated values. Red areas show either narrow hallways or a steep slope.
sis, but can bring human factors to the forethought of designers and architects, greatly influencing the style in which designs are created and simultaneously benefiting the building occupant. Recognizing the impossibility of designers to be a master of every field related to human ability, and the advantages of expert collaboration, a system has been presented in which biomechanics, architecture, and design can collaborate. Although some schools have been able to implement physical experiments to teach students Universal Design, many schools do not have the resources for these lengthy and costly experiments. Integrating human factor simulations in design programs would allow for students to learn the same basic principles.

ACKNOWLEDGEMENTS
I thank Sean Vance for allowing me to document his Universal Design Class, as well as Karl Daubmann and David Lantrip for their valuable feedback. I would also like to acknowledge the contribution of both Forest Darling and Robert Van Wesep for their help with programming and algorithms.

REFERENCES