Constraints for motion generation in work planning with digital human simulations

Michael Jonek[1][0000-0003-2489-6991], Tadele Belay Tuli[1][0000-0002-6769-0646] and Martin Manns[1][0000-0002-1027-4465]

[1] PROTECH-Institut für Produktionstechnik, Universität Siegen, Paul-Bonatz 9-11, 57076, Siegen, Germany

Abstract. Flexible and varied manual assembly processes in the automotive industry are often based on manual labor. While simulation can be used to improve planning to maximize efficiency while minimizing ergonomic issues for workers, common simulation tools require extensive modeling time. In such simulations, the users are often process engineers who want to easily create complex human motion simulations. This paper presents a concept developed to create complex human motions for interacting with objects in a production environment with little effort. The concept separates between geometric constraints and the semantic meaning of the respective geometry. With a set of data types developed for this purpose based on a unified ontology, a range of geometric and semantic information can be specified for arbitrary objects. In this way, an action-specific motion generator can be used to define the appropriate motion for the interaction with an object depending on the action without defining case-specific constraints. For a first proof, the concept is tested and demonstrated in the assembly of a pedal car and sitting on a chair at a manual workstation. Based on the use case, the effect of effort reduction is shown.

Keywords: Manufacturing planning, human motion simulation, manufacturing simulation, MOSIM, geometric constraints

1 Introduction

Digital modeling and simulation of human motions have become common tools in various industries over the last decades. Digital representation of reality enables predictions of real contexts. In many industries, this technology has now proven to be important for competitiveness (cf. [1]). In the automotive industry, simulations are used in e.g., production planning to evaluate buildability, ergonomics, task planning, timelines, and work safety. Similarly, in other industries, manual manufacturing engineers use simulations to develop manufacturing processes. In order to implement such simulations in small companies without large planning departments or access to simulation experts, reducing modeling effort is crucial [2, 3].

In this paper, the reduction of modeling effort for creating human motion simulation for assembly processes is considered with special focus on interaction between human avatar and objects. For these interactions, constraints are usually defined, which specify how the avatar should interact with objects such as tools or manufacturing products. Target constraints allow selection of an object as a target for reaching or walking motions. By selecting a point on the surface of an object, 3 or 6 degrees of freedom may be used to define restrictions on interaction motions. However, restricting to a specific point may lead to the risk of over-constraining and result in unrealistic poses if boundary conditions are not optimal, which in practice has to be constructed by time consuming manual constraint adjustments. Therefore, we address the question of whether there is a way to avoid the risk with relaxed constraints or not. Our approach is to constrain by describing an area of the object with multiple target possibilities within it. Furthermore, we propose concept of a standardized approach for constraint definition with low manual modeling effort.

To develop the concept, we search the literature for existing concept and consider them in relation to our use case. We then present our concept, explain the innovations compared to existing concepts, and apply it to our use case. Based on estimation data, we evaluate the effort involved in simulation generation by the number of constraints to be set.

2 Objective

The objective of the paper is to propose a generic concept to reduce the effort in creating human simulations for production planning by using a novel constraints concept. Using a manual workstation as an example, the concept is applied and demonstrated. The proposed concept is compared with existing concepts and the advantages and disadvantages are discussed.

3 Related Work

In recent years, digital human simulation (DHS) has been considered for representing an actual process in virtual environments e.g., for planning and simulating processes in the digital twin workplace. However, the inherent challenge in such an approach is to create plausible and realistic human postures or motions. In this context, approaches that are followed for defining constraints are crucial for improving simulation performance and planning efficiency.

Constraints that combine geometry and physical constraints have been described as useful for plausibility of motion and collision avoidance [4]. In manual assembly cases, constraints allow a flow of motion sequences and further manage process task planning [5]. Path constraints apply inverse kinematics to obtain full-body postures. For example, considering a grasping operation, the desired pose of a wrist joint will determine the remaining joints' pose [6]. However, such solutions require additional functionality for analyzing the feasibility of the solution from ergonomic and collision perspectives. In this regard, tools such as IPS-IMMA, EMA, and AnyBody are employed in various works [6–10]. For reachability and space requirements are the tool RAMSIS™ has been proposed [11].

In general, manual assembly process interactions comprise motions of objects and human workers. The interaction may be static, i.e., the digital human model (DHM) is constrained to object positions or dynamics i.e., with changes based on feasibility measures such as reachability. In this aspect, the interaction can be semantically described, for example, a top surface of a box, hold from both sides, pick object X and others. The semantic representations are also useful for associating behaviors with actions. In the MOSIM project, the semantic constraints are also used for describing the properties of scene components e.g. 3D scene object, required parameters, and motion types [12]. In [13], the interaction of a worker and an object has been represented using fuzzy constraints consisting of rules for assembly procedures. Latent space-based constraints have also been used for probabilistic motion generation techniques [14]. Considering assembly scenarios such as inserting a part, potential field models have been presented for path constraining during an assembly process [15].

In summary, constraint definitions presented in the literature are not simple and expressive. In some situations, it may cause functional redundancy and may lead to confusion e.g., selection between global and local coordinates. This may lead to over-constraining problem which may cause unrealistic postures. When it comes to implementation, simplicity, and efficiency in creating simulation setup are essential to improve lead-time. In this context, fast and straightforward constraint implementation should be investigated. In the existing solutions that are presented in this section, the constraints have to be set up individually for each scene. The parameters have to be adapted on a case-by-case basis in order for the constraint to work, which results in considerable modeling effort e.g. 10.5 – 21 h for less than 2 min of simulated time [16].

4 Our proposed constraints concept

The generic concept that we present involves a bilateral separation of the scene, which is helpful to reduce effort for large scenes. A library with standardized motions that are related to actions is set up separately from an object library. By separating actions from object classes semantics can be coupled depending on the context. Functional areas of space are defined using geometric primitives. These areas are semantically mapped to roles that depend on the type of activity. The space definition contains information about the spatial position and orientation. Thus, an action-specific motion generator can combine information from objects and actions for deriving information how the motion should be generated.

With an example calculation in Table 1, the effect of effort reduction with an action-based constraints concept compared to manually setting constraints can be shown. In the manual scene, the number of relevant constraints c must be set for each process instance a at each relevant object o. In our example scene, we consider 10,000 process instances (a = 10,000), for each of which 5 objects are relevant (o = 5), while for each object 2 constraints are relevant (c = 2). In total, 100,000 constraints must be set. For action-based constraint setting C is only required for one object type O and must be executed for each activity type A. The 10,000 process instances consist of 100 different activity types (A = 100) and 1,000 different object types (O = 1,000). For object types

5 constraints are defined to support different activities (C = 5). In this calculation, with 5,100 instead of 100,000, significantly less constraints have to be set.

Table 1 Comparison of the effort with an example calculation (n = number of constraint settings, A = activity types, a = process instances, C = constraints per object type, c = relevant constraints for object, O = object types, o = objects)

Manual setting of constraints	**Action-based setting of constraints**
$n = a \cdot o \cdot c$	$n = A + O \cdot C$

For example, consider a hollow tube which have to be manually inserted into another one. The object can have different functions, on which the constraints depend when interacting with the avatar. When the hollow tube is carried to the workstation, the handling is different than when it is inserted with force in a forward motion. Pedal car's front frame assembly simulation is one of the use cases where hand constraint can be applied. The wrist position is assigned on the surface of the object and fingers are aligned along the longitudinal axis of the hollow tubes, to make grasping easy (see Fig. 1 a).

To explain the constraint concept in detail, a chair is used, which can be used for sitting or leaning or is carried (see Fig. 1 b). It is also an object that is commonly found in production environments with manual workstations, demonstrating the scale effect in effort reduction. Depending on the interaction, an action-specific differentiation has to be made with respect to the setting of constraints. Depending on the interaction, the requirements for certain surfaces are defined. The seating surface can be constrained geometrically in three dimensions and with three orientation directions to define the target for a seating motion. The vector on the seating surface that points orthogonally upward is used as the target orientation for seating motion. It is further defined that the seat surface is the contact surface for the thighs. The definition works analogously for the backrest.

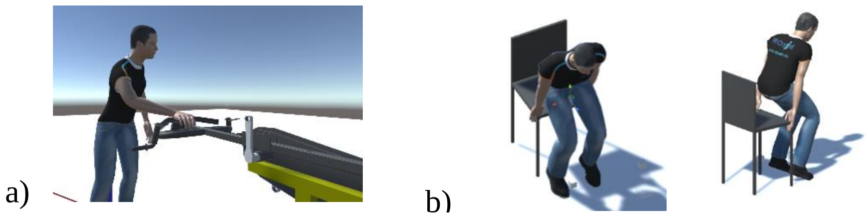

Fig. 1. a) Pedal car's front frame assembly simulation; b) Sitting on a chair simulation

In order to communicate the required information to an action-specific motion generator, a uniform ontology is chosen. For this purpose, a set of data types implemented in an Apache Thrift interface has been developed. The geometric constraints are based on primitive base geometries: Box, ellipsoid, and cylinder. The geometries can be scaled by intervals and restricted in rotation along the coordinate system. The constraints are defined in terms of a parent object and its local transformation in space. The object can move with the parent object. In order for the user or the object designer to be able to convey the functional meanings of the respective geometries, the generic semantic definitions of each parameter of the constraints must be implemented in their interface. For further elaboration, the functional constraint definition that has been implemented in MOSIM project for MMU development (cf. [12]) using Apache Thrift interface is demonstrated in Table 2.

Table 2 Constraint definition methods description

Description	**Constraint definition method**

Intervals between point	`struct MInterval {`
	`    1: required double Min;`
	`    2: required double Max;}`

Intervals in case of vector representation	`struct MInterval3 {`
	`    1: required MInterval X;`
	`    2: required MInterval Y;`
	`    3: required MInterval Z;}`

Geometric entity type	`enum MTranslationConstraintType {` `    BOX,ELLIPSOID,CYLINDER}`

The extent of translation and rotation	`struct MTranslationConstraint {`
	`    1: required MTranslationConstraintType Type;`
	`    2: required MInterval3 Limits;}`
	`struct MRotationConstraint {`
	`    1: required MInterval3 Limits;}`

Generic geometry constraint is defined as:	`struct MGeometryConstraint {`
	`    1: required string ParentObjectID;`
	`    2: optional MTransform ParentToConstraint;`
	`    3: optional MTranslationConstraint TrConstr;`
	`    4: optional MRotationConstraint RotConstr;`

Accordingly, a possible constraint solution for sitting on a chair is given in Table 3. In this geometry constraint, *ParentObjectID* is a required parameter that takes ID of an object in the scene graph. The object's base coordinate system serves as base for the constraint definition. Other parameters can be defined based on the desired motion behavior. If the transformation of an object with respect to the parent object is desired, *ParentToConstraint* can be applied. In some scenarios, when an offset from centroid is required, *TranslationConstraint* or *RotationConstraint* can be specified. The *WeightingFactor* is also an optional linear factor with values of 0 to 1, providing an extent of the constraint. In this aspect, the translational and rotational constraints are defined with intervals of a minimum and maximum values. The interval limits, for instance, allows the inverse kinematics to compute an optimal point within a defined a constrained space. The semantic description helps to describe further how to implement geometric constraint in simulation environment so that the end-user is able to understand and use it. Other types of constraints such as path, velocity, acceleration, joint, and posture can be defined in a similar manner.

Table 3 Implementation demonstration for geometry constraint of sitting on a chair

<table>
<tr><th>Description</th><th colspan="3">Functional representation</th></tr>
<tr><td>Interval for rotation in degree</td><td colspan="3"><code>new int_1 = MInterval(-90, 90)</code></td></tr>
<tr><td>Interval for translation in mm</td><td colspan="3"><code>new int_2 = MInterval(-10.0, 10,0)</code>
<code>new int_3 = MInterval(0.0, 0.0)</code></td></tr>
<tr><td>Transformation</td><td colspan="3"><code>trafo_1 = new MTransform("<ObjID>")</code></td></tr>
<tr><td>Box type geometry</td><td colspan="3"><code>new obj_type = MTranslationConstraintType.BOX</code></td></tr>
<tr><td>Rotation constraint</td><td colspan="3"><code>new int3_1 = MInterval3(int_3, int_3, int_1)</code></td></tr>
<tr><td>Translation constraint</td><td colspan="3"><code>new int3_2 = MInterval3(int_2, int_2, int_3)</code>
<code>new trl_1 = new MTranslationConstraint(</code>
<code>obj_type,int3_2)</code></td></tr>
<tr><td>Geometric constraint</td><td colspan="3"><code>new mGeoConst = MGeometryConstraint(</code>
<code>"<ObjID>", ParentToConstraint =</code>
<code>trafo_1, TranslationConstraint = trl_1,</code>
<code>RotationConstraint = int3_1,</code>
<code>WeightingFactor = 0.5)</code></td></tr>
<tr><th colspan="4">Semantic description</th></tr>
<tr><th>Name</th><th>Type</th><th>Required</th><th>Description</th></tr>
<tr><td>ObjectID</td><td>MGeometryConstr</td><td>yes</td><td>Object ID e.g., chair</td></tr>
<tr><td>Ground</td><td>MGeometryConstr</td><td>yes</td><td>Ground plane for foot</td></tr>
<tr><td>PosteriorPoint</td><td>MGeometryConstr</td><td>yes</td><td>Sitting surface area</td></tr>
<tr><td>DorsumPoint</td><td>MGeometryConstr</td><td>optional</td><td>Back rest surface area</td></tr>
</table>

5 Discussion

The presented concept has been successfully applied to the example of a chair. In terms of effort reduction, the calculation in Table 1 shows how the scaling effect works in comparison to existing concepts by comparing the number of attributes to be changed. Especially for large scenes with many repeating activity types and object types, the effect of effort reduction is significant. For the planning departments in companies, this results not only in a saving effort and time, but also in a reduction of the required knowledge. Experienced simulation experts are no longer necessary for each simulation, but only for the one-time predefinition of constraints for relevant objects. Simulations that are easier to use also enable smaller companies without large planning departments to plan production tasks, which often means that production processes can be designed more efficiently and ergonomically.

A consequence of the concept is that multiple solutions for target poses are possible. It therefore requires careful consideration of how large the intervals should be, since too large intervals can lead to unintended behavior, such as sitting on the edge of the target chair. However, the model has only been applied to two object types and would need to be evaluated with more objects. In the case of the object types demonstrated, there were multiple semantic meanings for which the model adds value. This is not the case for object types that are specialized for one application, such as screws. Also unresolved is the question of what about objects that can have many different applications, such as universal tools.

Currently, the concept is limited to a shop floor with known resources. Analogous to the two described objects, the concept can be used on most objects that are present in production. For an extension of the concept outside the production environment, further semantic information types can be added. In further research, an evaluation of the concept for an entire scene in manual manufacturing should be performed. To ensure that the concept is useful to the end user, research into the implementation of the concept in existing planning simulations is necessary.

6 Conclusion and outlook

In this paper, we present a new concept designed to create elaborate target constraints for production planning simulations with little end-user effort. By using the semantic description of objects, the concept enables the automatic setting of predefined constraints for each object. This allows simulations of larger scenes to be created with significantly less effort. In this way, economic benefits can be gained for companies by reducing the need for simulation experts and by generating simulations of larger scenes of production environments more efficiently. However, the concept has so far been limited to human simulations with objects in production environments and has been evaluated on few objects. For a successful application in production planning, further investigations on whole scenes with the concept would have to take place and an implementation of the concept in existing production simulations would have to be researched.

Acknowledgment

The authors would like to acknowledge the financial support by the Federal Ministry of Education and Research of Germany within the ITEA3 project MOSIM (grant number: 01IS18060AH).

References

1. Iriondo Pascual A, Högberg D, Syberfeldt A et al. Optimizing Ergonomics and Productivity by Connecting Digital Human Modeling and Production Flow Simulation Software. In: pp 193–204
2. Fritzsche L, Jendrusch R, Leidholdt W et al. (2011) Introducing ema (Editor for Manual Work Activities) – A New Tool for Enhancing Accuracy and Efficiency of Human Simulations in Digital Production Planning. In: Duffy VG (ed) Digital human modeling: Third international conference, ICDHM 2011, Orlando, FL, USA, July 9-14, 2011 ; proceedings. Springer, Heidelberg, pp 272–281
3. Jonek M, Manns M, Tuli TB (2021) Virtuelle Montageplanung mit Motion Capture Systemen/Virtual assembly planning with motion capture systems. wt 111:256–259. https://doi.org/10.37544/1436-4980-2021-04-78
4. Seth A, Vance JM, Oliver JH (09042007) Combining Geometric Constraints With Physics Modeling for Virtual Assembly Using SHARP. In: Volume 2: 27th Computers and Information in Engineering Conference, pp 1045–1055

5. Wan W, Harada K, Nagata K (2018) Assembly sequence planning for motion planning. AA 38:195–206. https://doi.org/10.1108/AA-01-2017-009
6. Björkenstam Staffan, Mårdberg Peter, Roller Michael et al. (2020) Digital Human Motion Planning of Operation Sequences Using Optimal Control of Hybrid Systems. Advances in Transdisciplinary Engineering 11:115–120. https://doi.org/10.3233/ATDE200016
7. Li Y, Delfs N, Mårdberg P et al. (2018) On motion planning for narrow-clearance assemblies using virtual manikins. Procedia CIRP 72:790–795. https://doi.org/10.1016/j.procir.2018.03.181
8. Illmann B, Fritzsche L, Leidholdt W et al. (2013) Application and Future Developments of EMA in Digital Production Planning and Ergonomics. In: Duffy VG (ed) Digital human modeling and applications in health, safety, ergonomics and risk management: 4th international conference, DHM 2013, held as part of HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013; proceedings. Springer, Berlin, pp 66–75
9. Fritzsche L, Ullmann S, Bauer S et al. (2019) Chapter 42 - Task-based digital human simulation with Editor for Manual work Activities – industrial applications in product design and production planning. In: Scataglini S, Paul G (eds) DHM and posturography. Academic Press, London, pp 569–575
10. Stambolian D, Eltoukhy M, Asfour S (2016) Development and validation of a three dimensional dynamic biomechanical lifting model for lower back evaluation for careful box placement. International Journal of Industrial Ergonomics 54:10–18. https://doi.org/10.1016/j.ergon.2015.12.005
11. van der Meulen P, Seidl A (2007) Ramsis – The Leading Cad Tool for Ergonomic Analysis of Vehicles. In: Hutchison D, Kanade T, Kittler J et al. (eds) Digital Human Modeling, vol 4561. Springer Berlin Heidelberg, pp 1008–1017
12. Gaisbauer F, Lehwald J, Agethen P et al. (2019 - 2019) Proposing a Co-simulation Model for Coupling Heterogeneous Character Animation Systems. In: Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, pp 65–76
13. Qu X, Nussbaum MA (2009) Simulating Human Lifting Motions Using Fuzzy-Logic Control. IEEE Trans Syst., Man, Cybern A 39:109–118. https://doi.org/10.1109/TSMCA.2008.2007996
14. Manns M, Fischer K, Du H et al. (2018) A new approach to plan manual assembly. International Journal of Computer Integrated Manufacturing 31:907–920. https://doi.org/10.1080/0951192X.2018.1466396
15. Al-Sultan KS, Aliyu MDS (1996) A new potential field-based algorithm for path planning. Journal of Intelligent and Robotic Systems 17:265–282. https://doi.org/10.1007/BF00339664
16. Manns M, Martín NAA (2013) Automated DHM Modeling for Integrated Alpha-Numeric and Geometric Assembly Planning. In: Abramovici M, Stark R (eds) Smart product engineering: Proceedings of the 23rd CIRP Design Conference, Bochum, Germany, March 11th - 13th, 2013. Springer, Berlin, pp 325–334