OntoBREP: An ontology for semantic boundary representations of geometry models

Alexander Perzylo, Junsheng Ding*

***Abstract*—This paper introduces an approach for leveraging CAD descriptions to a semantic level, in order to link additional knowledge to geometry models, interpret them in a larger context, and exploit resulting synergy effects. It uses a formal description language based on the Web Ontology Language (OWL), which is used to define boundary representations (BREP) of object geometries. This involves representing geometric entities in a semantically meaningful way, e.g., a circle is defined by a coordinate frame and a radius instead of a set of line segments. Furthermore, the scope of this semantic description language also covers geometric constraints and intended mechanical connections (semantic mates) between multiple objects. This allows deriving appropriate assembly poses from the type of connection and the geometric properties of the involved parts.**
This semantic representation is used to improve a variety of applications, ranging from automated geometric compliance checks to shape-based object recognition and constraint-based robot task descriptions.

I. Introduction

All aspects of industrial products are usually handled in a process called product lifecycle management (PLM). As part of this approach, the geometric product specifications are typically generated with Computer Aided Design (CAD) software. On the other side, a lot of applications are designed to use polygon-based geometry models, which only represent an approximation of the constructed model. This approach has many shortcomings, e.g., it only allows for a fixed level of detail and lacks the means to describe semantically meaningful geometries. For instance, a set of polygons which form a circular shape cannot easily be recognized as such, despite this information being already available at design time. If a circle and its parameters are described by the CAD model instead, the exact mathematical representation of the geometry is known and a triangulation can be carried out at any time using a level of detail that is most suitable to the current application's requirements. This separation of represented data and use case dependent calculations is as important as the separation of data and the software that was used to generate it. These paradigms allow flexible sharing and reuse of information.
Apart from polygon-based CAD formats, there are other formats that represent mathematical models of the contained geometries, e.g., STEP or IGES. STEP has been developed by the International Organization for Standardization as ISO 10303-21, whereas IGES was defined by the United States National Bureau of Standards as NBSIR 80-1978. Both formats have evolved to become widely accepted standards for CAD data exchange and storage. Still these formats lack the flexibility to be easily linked with other sources of information. Many pieces of CAD software feature the specification of materials for created geometries, but this information can only be exported to their proprietary data formats and is lost when a CAD model is exported to STEP or IGES.

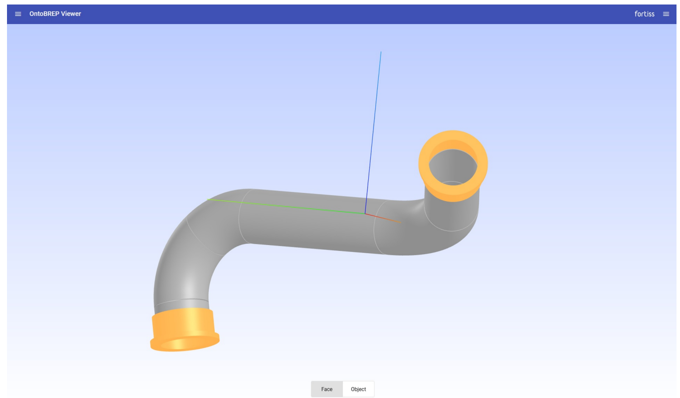


Fig. 1: Screenshot of OntoBREP Viewer used for inspecting OntoBREP geometry models loaded from a knowledge base, and to analyze them with the SPARQL query language.

In this paper we present a semantic description language for boundary representations of geometry models, i.e., a BREP ontology for points, curves, surfaces and volumes, using the Web Ontology Language (OWL). A complete specification of BREP's geometric and topological representation can be found in ISO 10303-42. Apart from the already mentioned benefits of having an open and application independent exchange format for CAD data, there are other major advantages. By having a semantic description language that is based on a logical formalism, it is possible to link to, and to automatically combine knowledge from different sources. For instance, a robot's CAD model can be extended by linking it with its kinematic model or a task specification can used geometric properties as a part of its parametrization.

Most of the contents of this workshop paper have already been published. In these previous papers, more information can be found on an example-guided introduction to the OntoBREP representation [1], an extension to OntoBREP for defining intended part connections that are used in an intuitive robot instruction framework [2], and the integration of OntoBREP-based object models in a knowledge-based manufacturing system [3].

*A. Perzylo and J. Ding are with fortiss, Research Institute of the Free State of Bavaria associated with Technical University of Munich, Munich, Germany. Please direct all correspondence to perzylo@fortiss.org

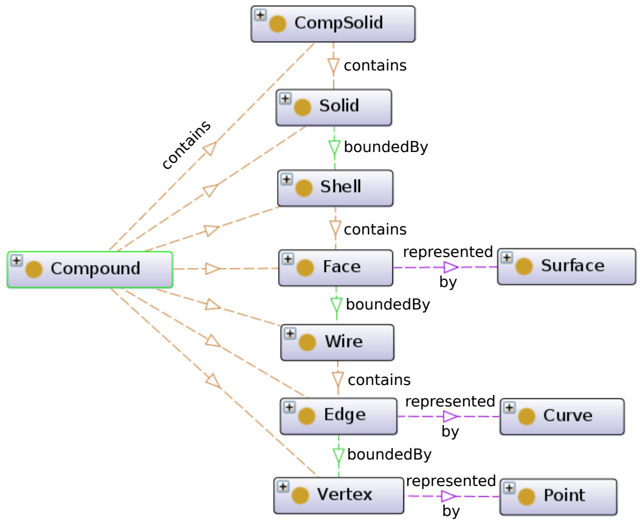


Fig. 2: Overview of the OntoBREP structure (class-level). The knowledge model contains topological entities (left and center) and geometric entities (right).

II. Related Work

There have been different approaches for enriching CAD models with semantics. OntoCAD [4] is an ontological annotation approach that is based on labeling geometry elements of CAD models using concepts from a CAD ontology. [5] created an ontology for describing CAD models in the Drawing Exchange Format (DXF). In [6], the authors presented an ontology-based approach of semantically describing CAD data and features. They use a rule system to automatically classify CAD features in order to provide a compatible mapping between different CAD systems. [7] showed how not only CAD models but the complete Product Lifecycle Management (PLM) approach can be semantically modeled.

The RoboEarth project [8] followed a holistic approach to knowledge representation and sharing for robots. The RoboEarth language [9], which was built using the Web Ontology Language (OWL), was used to describe many aspects of robot tasks, e.g., task structures, semantic environment maps and models of interaction objects. RoboEarth object models are, however, only meta-descriptions for linked industry-standard CAD models.

[10], [11], [12] presented an approach for describing coordinate frames and geometric constraints between the frames for applications in the robotics domain.

In this paper, we introduce a semantic description language to describe not only meta-information for a given CAD model [9], but also the content of the model. Given the complete ontological representation of CAD models, we can define geometric constraints not only on the object level [9] or only on a frame level [11], but also on all semantically meaningful intermediate levels, e.g., points, edges, and faces of CAD models.

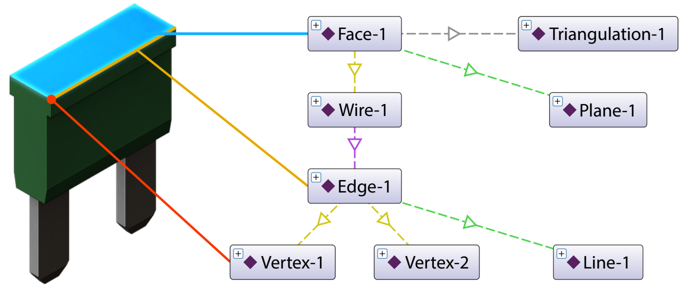


Fig. 3: Excerpt from a semantic model of a blade fuse's geometry based on the OntoBREP ontology (instance-level).

III. OntoBREP Concept

A Boundary Representation (BREP) of CAD models describes the geometric properties of points, curves, surfaces and volumes based on their exact mathematical models. Such CAD models are created by combining given base geometries and corresponding boundary limits. The BREP specification distinguishes geometric and topological entities, as illustrated in Fig. 2. Geometric entities hold the numerical data, while the topological entities arrange them in a hierarchical fashion. Following this structure, an ontology for describing topological and geometric entities and their relations has been developed in the the Web Ontology Language (OWL) [1].

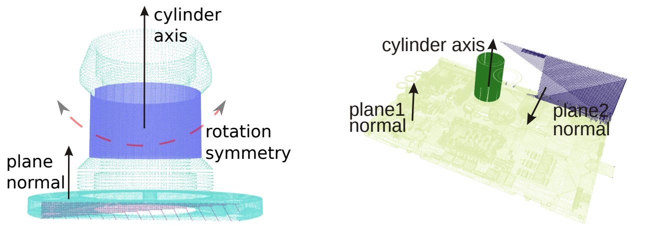
A. Topological Entities

The BREP standard specifies eight kinds of topological entities: *Vertex*, *Edge*, *Face*, *Wire*, *Shell*, *Solid*, *CompSolid*, and *Compound*. Only *Vertices*, *Edges*, and *Faces* have direct links to geometric entities. A *Vertex* is represented by a point. An *Edge* is represented by a curve and bounded by up to two *Vertices*. A *Wire* is a set of adjacent *Edges*. When the *Edges* of a *Wire* form a loop, the *Wire* is considered to be closed. A *Face* is represented by a surface and bounded by a closed *Wire*. A *Shell* is a set of adjacent *Faces*. When the *Faces* of a *Shell* form a closed volume, the *Shell* can be used to define a *Solid*. *Solids* that share common *Faces* can be grouped further into *CompSolids*. *Compounds* are top-level containers and may contain any other topological entity.

B. Geometric Entities

The topological entities may link to three types of geometric entities, which are *Points*, *Curves*, and *Surfaces*. They represent 0-, 1-, and 2-dimensional geometries respectively. *Curves* and *Surfaces* are defined through parametrizable mathematical models. Supported curve types can be categorized as unbounded curves, e.g., lines, parabolas, or hyperbolas, and bounded curves, e.g., Bezier curves, B-spline curves, circles, or ellipses. Offset curves represent a translated version of a given base curve along a certain vector, whereas trimmed curves bound a given base curve by limiting the minimum and maximum parameters of their mathematical model. In case the

[1] https://github.com/OntoBREP/ontobrep



(a) Recognition of underspecified pose for symmetrical object

(b) Estimation of fully specified pose using primitive shapes

Fig. 4: Estimation of object poses using primitive shape constraints based on CAD semantics

exact model is unknown, a curve might also be approximated by a polygon on triangulated data. The geometric representation of an *Edge* might be specified by a 3D curve, or a 2D curve in the parameter space of each surface that the *Edge* belongs to.
Surfaces rely on unbounded mathematical models, e.g., planes, cones, or cylindrical surfaces, and bounded models, e.g., Bezier surfaces, B-spline surfaces, spheres, or tori. Surfaces can also be defined as linearly extruded curves. An offset surface translates a base surface along a given vector, and a revolution surface is created by rotating a given base curve around a given direction vector.

C. Example OntoBREP Model

Fig. 3 visualizes an excerpt of a geometry representation based on the OntoBREP ontology. The top face (Face-1) of the depicted automotive blade fuse's geometry is defined by an infinite plane (Plane-1). The individual Wire-1 is a topological BREP entity that contains a total of four edges that make Face-1 a finite rectangular-shaped surface. The highlighted edge (Edge-1) is defined by a line (Line-1) bounded by two vertices (Vertex-1 and Vertex-2). In addition to this boundary representation, the individual Triangulation-1 provides a triangulation of Face-1 containing a set of triangles that covers its surface. The triangulation data can be used for visualization purposes, among others.

D. OntoBREP Converter Tool

In order to mitigate the required efforts for describing a CAD model using our semantic description language, we implemented a software component for converting STEP and IGES files to the proposed ontology-based representation. As almost all widely used CAD software support these interchange formats, our converter enables us to continue relying on these mature and highly specialised tools for designing the CAD models. The converter has been implemented in C++ and uses the community edition of Open CASCADE [2] for parsing STEP and IGES files and creating triangulations that are used in our OntoBREP viewer 1.
For a more direct generation of OntoBREP models, we have additionally developed a Solidworks plugin that allows to directly export SolidWorks models from the tool's GUI.

IV. Applications

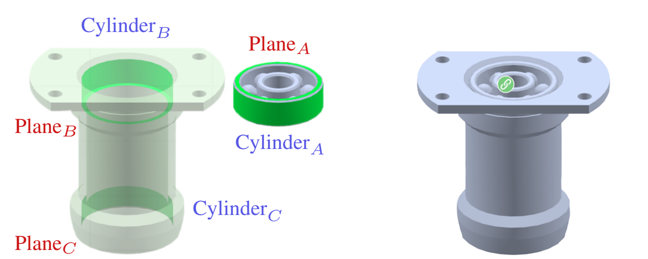
In the following subsections, application examples are presented that have been realized with the help of OntoBREP models. For all experiments, a central knowledge base (Ontotext's graph database GraphDB) was used to persistently store the required models.

A. Automatic Analysis of Geometries

Based on the OntoBREP representation, all geometric specifications of interaction objects as well as environment descriptions can be stored in the knowledge base, where they are subject to automatic reasoning mechanisms that may automatically infer implicit insights from explicitly modelled facts. Through the SPARQL-enabled REST interface, all of these models can be analysed and linked with additional pieces of information, e.g., regarding an industrial manufacturing process.
For instance, given a geometry model of a pipe system, a SPARQL query can be designed to identify the types and locations of connection flanges between individual pipe segments. For highlighting purposes, a query may retrieve all Face individuals that are part of the identified subgeometries 1. Such findings can then be automatically used as annotations for the models. Listing 1 shows a simplified query for retrieving the 6D poses of all flanges of pipe segment *http://www.fortiss.org/ont/FL3_SOI_Rohrleitung.ttl*.

B. Object Recognition

Object recognition and pose estimation using CAD models are well-known research area in the field of computer vision. There are several approaches that rely on object detection models composed of primitive shapes. We extended our previous work [13] to directly obtain primitive shapes from object models that are described using the BREP ontology presented in this paper. This improves the performance by eliminating



(a) Two solutions to intended part connection of ball bearing

(b) Resulting assembly pose after solving constraints

CylinderCylinderConcentricConstraint(Cylinder$_A$, Cylinder$_B$)
PlanePlaneCoincidentConstraint(Plane$_A$, Plane$_B$)

Fig. 5: Geometric constraints that specify an intended part connection that can be used to define an assembly task involving two objects

[2] http://www.opencascade.org/

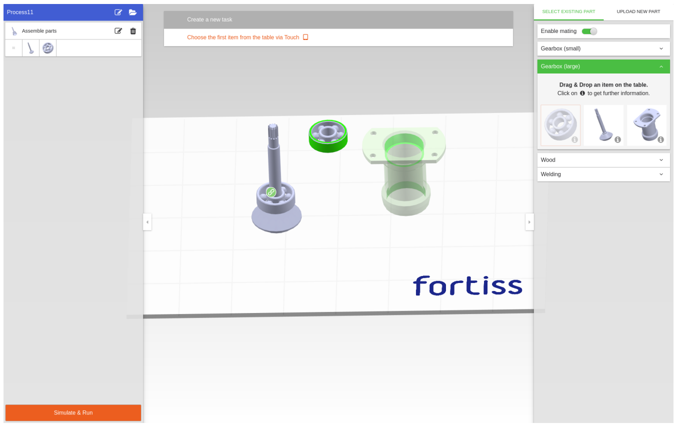


Fig. 6: Graphical user interface for defining robotic assembly tasks. Annotated OntoBREP models are used to automatically identify suitable assembly poses (highlighted in dark green).

errors due to incorrect detections of shape primitives from point cloud models.
In [14], we presented an approach for detecting symmetrical objects using geometric constraints. This was extended to use semantic BREP models of primitive shapes and geometric constraints between them (see Fig. 4). The resulting under-specified object poses, controlled-spaces, and null-spaces can now be represented as geometric entities themselves.

C. Task specification based on geometric constraints and high-level definitions of intended part connections

Intuitive interfaces for robot programming [15] are an important target application of this work. Semantically rich descriptions of CAD models allow users to refer not only to manipulation objects, but to the geometric entities that they are comprised of. Robot tasks can then be defined using interrelation constraints between the relevant geometric entities, e.g., to specify an assembly pose [16]. Fig. 6 shows a dialog of a graphical user interface used to visualize OntoBREP models for the purpose of defining assembly tasks. Through a reusable and OntoBREP-based semantic modelling of intended part connections, e.g., describing that a ball bearing's meaningful assembly pose is defined by two geometric constraints: a plane-plane-coincident constraint and a cylinder-cylinder-concentric constraint (see Fig. 5), the task of defining these geometric constraints for a specific assembly can be reduced to simple drag-and-drop operations [2]. Combined with object poses obtained from a vision system (see Section IV-B), the geometric constraints are solved to generate target poses for task execution.

D. Constraint-based robot control

In this application, the robot task specifications that are created using geometric constraints are combined with semantic descriptions of the robotic workcell. A semantic workcell description contains the workcell layout, sensors, tools, robots and their kinematic structures, etc. The task's geometric constraints are solved to generate corresponding constraints on the

```
PREFIX cad: <http://www.fortiss.org/ont/ontobrep#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?pos_x ?pos_y ?pos_z ?vecN_x ?vecN_y ?vecN_z
WHERE {
  VALUES ($object)
   {(<http://www.fortiss.org/ont/FL3_SOI_Rohrleitung.ttl#Compound1>)}
   ?object cad:contains ?solid . ?solid cad:boundedBy ?shell .
   ?shell cad:contains ?face .
   ?face cad:representedBy ?surface .
   ?surface rdf:type cad:Plane .
   ?face cad:boundedBy ?oWire .
   ?oWire cad:firstElement ?oEdge .
   ?oEdge cad:representedBy ?oCurve .
   ?oCurve rdf:type cad:Circle .
   ?oCurve cad:radius ?oRadius .
   ?face cad:contains ?iWire .
   ?iWire rdf:type cad:Wire .
   ?iWire cad:firstElement ?iEdge .
   ?iEdge cad:representedBy ?iCurve .
   ?iCurve rdf:type cad:Circle .
   ?iCurve cad:radius ?iRadius .
   FILTER (?oRadius/?iRadius > 1.3) .
   ?surface cad:position ?pos .
   ?pos cad:x ?pos_x . ?pos cad:y ?pos_y . ?pos cad:z ?pos_z .
   ?surface cad:directionNormal ?vecN .
   ?vecN cad:x ?vecN_x . ?vecN cad:y ?vecN_y . ?vecN cad:z ?vecN_z .
}
```

Listing 1: SPARQL query returning the positions and orientations of all flanges of given pipe segment. Identification of flanges is based on a given minimum ratio of inner and outer diameters of circular boundaries of planar surfaces.

robot's pose. The resulting robot-dependent null-space of the solved geometric constraints can be linked to the task instance. The constrained robot poses are then executed on the low-level robot controller, which may exploit the information about null-spaces to optimize the robot's motion without violating the given constraints. Details of this application can be found in [17].

V. Conclusion

This paper presents a semantic description language for CAD models that is based on a boundary representation of points, curves, surfaces, and volumes. This language also offers features to specify geometric constraints between arbitrary parts of CAD models They may not only refer to the coordinate frame of an object, but also directly to its vertices, edges, or faces. The benefits of this approach have been showcased through a set of applications, which range from automated geometric compliance checks to shape-based object recognition and intuitive parametrization of task descriptions. By grouping geometric constraints into higher-level semantically described part connections, the approach can be used to simplify the instruction of robotic assembly tasks. The combination of the proposed geometry models with rich semantic descriptions of processes, additional information on products, and manufacturing resources enables a cognitive robot system to interpret all of these linked models in order to automatically generate corresponding manufacturing plans and assess the current context of the task at hand.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952197.

REFERENCES

[1] A. Perzylo, N. Somani, M. Rickert, and A. Knoll, "An ontology for CAD data and geometric constraints as a link between product models and semantic robot task descriptions," in *Proc IEEE Int. Conf. Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015, pp. 4197–4203.

[2] F. Wildgrube, A. Perzylo, M. Rickert, and A. Knoll, "Semantic mates: Intuitive geometric constraints for efficient assembly specifications," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6180–6187.

[3] A. Perzylo, I. Kessler, S. Profanter, and M. Rickert, "Toward a knowledge-based data backbone for seamless digital engineering in smart factories," in *Proceedings of the IEEE International Conference on Emerging Technologies And Factory Automation (ETFA)*, 2020.

[4] C. Li, C. McMahon, and L. Newnes, "Progress with OntoCAD: A standardised ontological annotation approach to CAD systems," in *International Conference on Product Lifecycle Management (PLM)*, Eindhoven, Netherlands, July 2011.

[5] L. E. R. Garcia, "Ontological CAD data interoperability framework," in *4th International Conference on Advances in Semantic Processing (SEMAPRO)*, Florence, Italy, Oct. 2010.

[6] S. Tessier and Y. Wang, "Ontology-based feature mapping and verification between CAD systems." *Advanced Engineering Informatics*, vol. 27, no. 1, pp. 76–92, 2013.

[7] A. Matsokis and D. Kiritsis, "An ontology-based approach for product lifecycle management," *Computers in Industry*, vol. 61, no. 8, pp. 787–797, Oct. 2010.

[8] M. Waibel, M. Beetz, J. Civera, R. D'Andrea, J. Elfring, D. Gálvez-López, K. Häussermann, R. Janssen, J. Montiel, A. Perzylo, B. Schießle, M. Tenorth, O. Zweigle, and R. van de Molengraft, "Roboearth - a world wide web for robots," *IEEE Robotics and Automation Magazine (Special Issue Towards a WWW for Robots)*, vol. 18, no. 2, pp. 69–82, 2011.

[9] M. Tenorth, A. Perzylo, R. Lafrenz, and M. Beetz, "Representation and exchange of knowledge about actions, objects, and environments in the roboearth framework," *IEEE Transactions on Automation Science and Engineering (T-ASE)*, vol. 10, no. 3, pp. 643–651, 2013.

[10] J. De Schutter, T. De Laet, J. Rutgeerts, W. Decré, R. Smits, E. Aertbeliën, K. Claes, and H. Bruyninckx, "Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty," *The International Journal of Robotics Research*, vol. 26, no. 5, pp. 433–455, 2007.

[11] T. De Laet, S. Bellens, R. Smits, E. Aertbelien, H. Bruyninckx, and J. De Schutter, "Geometric relations between rigid bodies (part 1): Semantics for standardization," *IEEE Robotics Automation Magazine*, vol. 20, no. 1, pp. 84–93, Mar. 2013.

[12] T. De Laet, S. Bellens, H. Bruyninckx, and J. De Schutter, "Geometric relations between rigid bodies (part 2): From semantics to software," *IEEE Robotics Automation Magazine*, vol. 20, no. 2, pp. 91–102, June 2013.

[13] N. Somani, E. Dean-Leon, C. Cai, and A. Knoll, "Scene perception and recognition in industrial environments for human-robot interaction," in *International Symposium on Visual Computing (ISVC)*, July 2013.

[14] N. Somani, C. Cai, A. Perzylo, M. Rickert, and A. Knoll, "Object recognition using constraints from primitive shape matching," in *International Symposium on Visual Computing (ISVC)*, Dec. 2014.

[15] S. Profanter, A. Perzylo, N. Somani, M. Rickert, and A. Knoll, "Analysis and semantic modeling of modality preferences in industrial human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2015.

[16] A. Perzylo, N. Somani, S. Profanter, M. Rickert, and A. Knoll, "Toward efficient robot teach-in and semantic process descriptions for small lot sizes," in *Robotics: Science and Systems (RSS), Workshop on Combining AI Reasoning and Cognitive Science with Robotics*, Rome, Italy, July 2015.

[17] N. Somani, A. Gaschler, M. Rickert, A. Perzylo, and A. Knoll, "Constraint-based task programming with CAD semantics: from intuitive specification to real-time control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2015.