DESIGN RULES WITH MODULARITY FOR ADDITIVE MANUFACTURING

H. Jee*†, Y. Lu†, and P. Witherell†

*Mechanical System and Design Engineering, Hong Ik University (HIU),
Seoul 121-791, South Korea
†Systems Integration Division, National Institute of Standards and Technology (NIST),
Gaithersburg, MD 20899, U.S.A.

REVIEWED

ABSTRACT

Design rules for additive manufacturing (AM) can help ensure manufacturability, which can be viewed as compatibility between designs and the fabrication processes that produce those designs. Additionally, such rules frequently provide direct guidelines or constraints for designing AM-destined parts. Here, we present design rules as sets of modular components and associated formalisms. Independent of context, these representations can be more easily interpreted and efficiently implemented. Given context, components are specialized to represent process-specific parameters for different AM builds and processes. This method of specialization enables designers to reconfigure design rules, rather than create new rules from scratch, thus preserving fundamental AM principles while supporting customization and explicit representation.

1 INTRODUCTION

A rule can be defined as “a prescribed guide, a valid generalization, or a standard of judgment” [1] for conduct or action. Design rules, therefore, often provide direct guidelines or constraints in relation to part designs, process parameters, and material properties. Design rules also provide a means to control costs, time, and outcomes for both the design and manufacturing phases of the product realization process [2]. This control allows a designer to verify correctness in designs prior to manufacturing. For instance, in semiconductor manufacturing [3], ‘design-rule checks’ (DRC) help manage various complexities associated with 1) a circuit layout, including geometric representations and 2) data during the manufacture of a working design [4]. Over time, design rules often become a basic requisite for many innovation-driven fields. One such field is additive manufacturing (AM).

AM technologies provide tremendous flexibility for designers because of the wide range of complex geometries that they can produce. For this reason, design rules in AM are desired because they can provide much needed insight into manufacturability for a particular material, design, or process. That insight leads to understanding direct guidelines or constraints during process planning. Design rules can be critical to satisfying manufacturability because they enable designers to determine the best process and material combinations for a design.

The aforementioned criticality is based on how mechanical properties are determined in a final part. In AM, mechanical properties of a manufactured part are as dependent on the process specifics as they are on the raw material. To compound challenges, process/material relationships are not consistent and will differ based on both the material and process type. ASTM F2792 has defined seven different categories of AM processes that support layer-by-layer fabrication [5]. Variations between these processes and materials significantly compound design spaces – going way beyond complications created by basic geometries. Design rules provide a means to manage complications created by those variations. Design rules are capable of limiting design features for different process/material/parameter combinations. A given set of design rules can effectively constrain a design space. However, members of this "given set" are subject to the specifics of AM materials and processes.

Variations in AM builds are created from multiple different sources; these sources include materials, processes, equipment manufacturers, and machines. In AM, design rules are abstracted from observations and correlations made between material, process, and design. Design rules derived from one data source will likely differ from design rules that are derived from other data sources. In addition to inconsistencies created by data sources, ambiguities can arise from the way the design rules are communicated and implemented. Different language and syntax can influence how design rules are constructed and interpreted. For reasons such as these, AM design rules can become so generalized that their value to the designer becomes diluted. In this paper, we propose a formal approach to design-rule representation to manage potential ambiguities.

Here, we propose the adoption of principles and formalisms that allow us to modify, extend, reconfigure, or customize generalized rules as needed - instinctively and deliberately. Formalisms provide both structure for the generalizations and a means to tailor that structure for a specific process, machine, or build. Overall, this approach will 1) promote the consistent application of design rule principles, 2) mitigate ambiguities and inconsistencies that may be introduced both within and between processes, 3) provide guidelines for the generation of new design rules, and 4) establish design rules as sets of modular components and associated formalisms.

In the following sections, we first review sources of variations in design rules, where dilution and ambiguities can diminish their effectiveness. We then introduce our approach for incorporating formalisms into design rules. In the final stage of the paper, we will demonstrate as a case study how process-independent representations provide a basis for specific interpretations of design rules for an AM process.

2 BACKGROUND AND APPROACH

Process-dependent guidelines and design rules have been widely developed for AM as "further process specific restrictions, explored and summarized in a simple and intuitive way, will offer a comprehensive overview of its limitations and possibilities [6]." Many researchers have focused on developing, and thus providing, prescriptive guidelines or explicit constraints for AM-destined designs, mostly for either the process categories of '*material extrusion* (ME)' [7,8,9] or '*powder bed fusion* (PBF)' [10,11,12,13,14,15] as defined in ASTM F2792 [5].

The development of generalized design rules to accommodate process variations has also been a focus of current research. For instance, Adam and Zimmera proposed a generalized approach for handling geometry issues in developing compatible design rules over different AM processes. First, they implemented comprehensive design rules by defining process-independent, geometrical, and standard elements. Next, they developed parametric templates compatible with most AM processes. Then, they used elements and templates to develop process-specific design rules based on independent functions [16]. Gibson et al., on the other hand, developed and validated a finite element model of a viscoelastic feature in materials. This model can be used to 1) provide sets of rules defining lower limits of designs and 2) create reliable designs for a given specific AM process [17]. Many other approaches can be found in [18,19,20].

From a knowledge-based view, design rules constitute a body of knowledge (BoK) that contains information about allowable design actions. This BoK could become "a common source of a vast amount of information collected, compiled, and established by collaboration between industry and academia [13]." Therein lays the challenge! How do we provide a common set of instructions to the users of manufacturing processes that are inherently different?

2.1 An "Overhang" Example

Several types of AM processes are unable to build material over spatial regions without proper support. For these processes, the unsupported material, or 'overhang,' must be carefully managed to preserve the integrity of the design. Management of overhangs is a common example in a set of generalized design rules. Before fabrication using particular processes, support structures are often necessary to prevent the overhang from failing. The design rule should leave no doubt when supports are necessary to build overhangs (unless overhangs can be minimized due to changes in the build orientation [21, 22]).

The rule stated above has variations. For example, according to investigations by Thomas [10], rules provided for overhangs can be in the form of 'self-supporting radii or holes', as in the case of self-supporting angles, or vary depending on the type of overhang, material, and process. Exact examples of such guidelines are listed below.

- "The lowest angle of a flat downward-facing overhang turned out to be 45 degrees [10]" (PBF process).
- "The smallest radius of self-supporting curve required a 28 degrees tangent on the radii and the largest one a 40 degrees tangent [10]" (PBF process).
- "The smallest holes of the self-supporting type appeared to be the least accurate at 0.3mm and holes at 5mm radius and above were all within a tolerance of 0.1mm [10]" (PBF process).
- "The common overhang distance for all layer thickness is approximately 0.075mm to 0.08mm [10]" (PBF process).
- "Self-supporting angle varies depending on the material, but is usually around 45 degrees [8]" (ME process).
- "The de-facto value is 45 but most printers with some active cooling can handle a bit more. Try 55 and decrease if there is unwanted drooping, curling, or noodling' [9]" (ME process).
- "Angled surfaces (30~45): self supporting with rough surface finish [14]" (PBF process).

The above variations of guidelines for overhangs are now re-written as structured rules, with antecedents and consequents:

- Overhangs (angular), if designed at greater than 45 degrees of undercut angle and built by a PBF process, are self-supporting.

- Overhangs (circular), if designed at proper range of tangent angles between 28 and 40 degrees corresponding to undercut radii and built by a PBF process, are self-supporting.
- Overhangs (hole), if modified with a peak under a proper range of undercut angles and built by a PBF process, are self-supporting.
- Overhangs (angular), if designed at less than around 0.075mm offset layering and built by a PBF process, are self-supporting.
- Overhangs (angular), if designed at greater than around 45 degrees of undercut angle and built by an ME process, are self-supporting.
- Overhangs (angular), if designed at greater than around 55 degrees of undercut angle, accompanied with unwanted drooping, curling, or noodling and built by an ME process, are self-supporting.
- Overhangs (angular), if designed at greater than around 30 degrees of undercut angle, aided by rough surface finish and built by a PBF process, are self-supporting.

2.2 Modularizing Representation

The rules presented in Section 2.1 can be re-written and represented using one single descriptive expression roughly formalized as:

Category (type), if {conditions} then {consequences};

where 'category,' 'conditions,' and 'consequences' include two components: *primitives* and *modules*.

A *primitive* (*or measured primitive*) is a feature parameter such as 'undercut angle,' 'undercut radius,' 'overhang distance,' and 'raw material type.' *Module* is an implicit design feature such as 'overhang,' 'support structure,' and 'surface finish.' Accordingly, and more importantly, both *primitives* and *modules* are process-independent. This means that design guidelines and rules that are written as prescriptive guidelines or explicit constraints can be re-established into design rules using only *primitives* and *modules*. In doing so, design rules can be interpreted more explicitly and implemented more efficiently because they are independent of context. Thereafter, they can be dynamically reconfigured, rather than created from scratch, from individual components for different AM builds and processes.

Here, we propose a methodical approach to provide such modularity in design rules. Our approach is based on developing and introducing sets of fundamental building blocks. Such sets will be based on the following three premises:

- First, design rules should be established as process-independent sets of modular components. For a specific AM build and process, these components can be specialized with process-specific parameters.
- Second, in process-specific implementations, design rules should be reconfigured and repurposed from existing discrete components, rather than created from scratch.
- Third, the principles abstracted from design rules should be robust enough to be impervious to fundamental changes or evolutions in layer-by-layer processes [23].

A more specific explanation of these principles, and their use in a case study, will be provided in the following sections.

3 THE METHODOLOGY

Design rules constitute a BoK that contains information useful for AM designers. In our view, the elements of design rules represent guidelines and constraints that make explicit how design and process information is managed, irrespective of the exact content. New design rules may be desired to effectively represent a new process, to incorporate new knowledge, or to develop in-house applications.

Our methodology for design rule representation supports reusability, extendibility, readability, and computability, as shown in Fig. 1. As noted above, representations consist of well-defined components, consisting of two model classifications: *primitives* and *modules*. *Primitives* represent fundamental concepts of physical parameters observed in AM; *modules* establish dependencies between *primitives* and between *primitives* and other *modules*. Implementation-specific design rules can then be developed by

- Identifying the process-independent modular components
- Defining the process-specific dependencies between modular components based on observed behavior
- Assigning values to modular components as appropriate

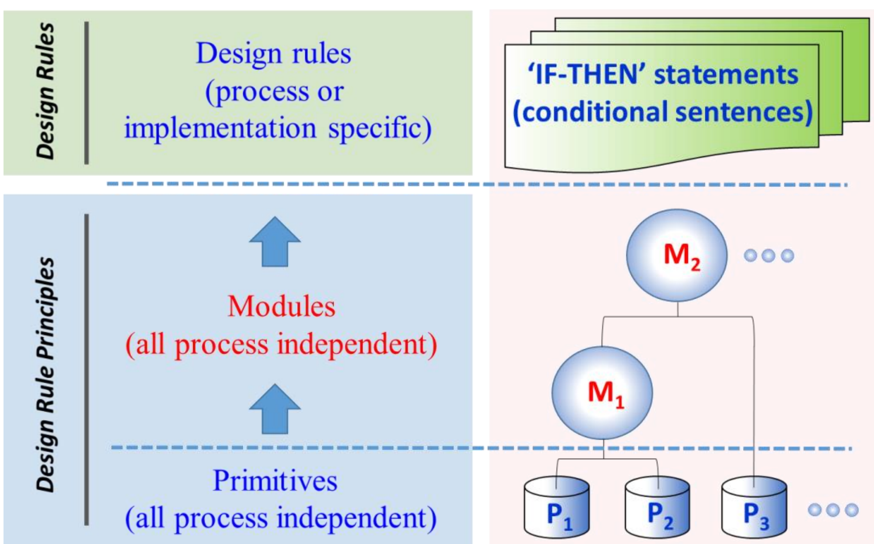

Fig. 1 Design rule principles and design rules with modularity for AM

Design rules are expressions and sentences composed of different language elements such as words, phrases, clauses, and sentences. Different languages and syntaxes can lead to variations in how design rules are constructed and interpreted. Design rules, written as prescriptive guidelines or explicit constraints specific to an AM-destined design [6-15], are more often than not dependent on a specific process. A transformation to process-independency will result in generalized, modular expressions. This section describes both the principles and the formalisms needed to execute such a transformation.

First, proper language elements in design rule expressions are re-defined into modular components and generalized as process-independent. The main purpose of design rules is to realize the 'design allowable.' In general, a design rule is associated with two semantic notions: one is 'designs (designed geometric features)' implicitly generated or developed, and the other 'guidelines or constraints' explicitly defined or designated. For this reason, language elements in a design rule expression or representation fall, literally and semantically, into a category either of designs or of guidelines or constraints. Thereafter they are classified into two components depending on the connotation they intend to carry; *module* conceiving implicit designs and *primitive* conceiving explicit guidelines or constraints. Table 1 shows some of the key attributes we considered in determining *modules* and *primitives*.

Table 1. Types of attribute and their comparison of modular components

Type of attribute	*module*	*primitive*
Semantic connotation	implicit (designs)	explicit (guidelines or constraints)
Methods of generation	derived or developed	measured or defined
Dependency	yes	no

Second, design rules are re-written using conditional sentences, which provide descriptive expressions that form the basis for formalization. They include both the 'conditions,' called hypothetical situations, and the 'consequences.' A full conditional sentence using a variety of grammatical forms and constructions generally contains two clauses: "the dependent clause expressing the conditions and the main clause expressing the consequences [24]." Here, an expression of design rules is formalized as a collection of 'IF-THEN statements,' as "the dependent clause is most commonly introduced by the conjunction 'if' [25]." It is written mathematically as $p \rightarrow q$; $\textit{if} \;\; p \; \textit{then} \; q$ where p is the dependent clause expressing the conditions and q is the main clause expressing the consequence. These two clauses, p and q, consist of two modular components, *primitives* and *modules,* noted above. This means that design guidelines and rules that are written as prescriptive guidelines or explicit constraints can be re-established into design rules with modularity using only *primitives* and *modules*.

The information needed to define *primitives, modules*, and their representations as conditional sentences are embedded in existing design rules, of which there are many [6-15]. That information is 1) embedded in the prescriptive guidelines or constraints contained in those rules and 2) represented with a number of different formalisms, including unstructured English. Currently, our approach relies on intuitively abstracting information from available guidelines and rules. Based on our interpretations, we then organize that information into *primitives* and *modules* – as defined below. As we continue to develop our methodology, we will explore more deterministic methods for correlating *primitives* and *modules*.

3.1 *Primitives*

A *primitive* - independent in nature, measured or defined as designated, - is an explicit, single, identifiable entity. *Primitives*, however, are not necessarily interpreted in isolation since they are used to calculate a measure or designate a parameter in an AM process. Conceptually, *primitives* are "a set of standardized parts or independent units that can be used to construct a more complex structure [26]." This means that a *primitive* can exist with little or no customization at all. According to ASTM F2792, all seven categories of AM machines need a discretized version of the computer-aided design (CAD) geometry of a part that supports a layer-by-layer fabrication [5]. In addition, six among those seven need a discretized version of every single layer.

Primitives can be classified into three groups: geometry, processes, and materials, depending on their source of generation. Several examples of each category are given in Table 2. These three categories of *primitives* are common to each of the seven ASTM F2792 categories of AM processes. As such, they are process-independent. They become process-dependent when specific values and ranges are acquired from direct observation of those processes. Both values and ranges must be verified before these *primitives* can be used to develop process-specific design rules.

Table 2. Three different categories of *primitives* common to different AM processes

Geometry	Process	Material
○ feature dimensions ○ feature locations ○ feature orientations ○ undercut angle	○ layer thickness ○ scan resolutions ○ build speed ○ build power ○ build orientations ○ platform dimensions	○ raw material type ○ material density

3.2 *Modules*

A *module* - or 'context-to-*primitive*,' derived or developed as dependent - is an implicit feature (often geometric), that is acquired from indirect observation (e.g., function) of given sets of either *primitives* or *modules*, or both. We have developed three types of *modules*.

Type I: A *module* can be composed only from a set of *primitives*, for example:
- A *module* for an overhang rule could be derived from two associated *primitives*, feature dimension and undercut angle, each of which has a certain range of values.
- A *module* for a surface finish rule could be derived from three associated *primitives*, layer thickness, scan resolutions, and raw material type, each of which has a certain range of values or types.
- A *module* for a porous part rule could be derived from three associated *primitives*, layer thickness, scan resolutions, and raw material type, each of which has a certain range of values or types.

Type II: A *module* can be composed from both *primitives* and other pre-developed *modules*, for example:
- A *module* for a lattice structure rule could be developed from another pre-developed *module*, porous part, plus three associated *primitives,* feature dimension, feature locations, and feature orientations, each of which has a certain range of values.

Type III: A *module* can be derived only from other pre-developed *modules*, for example:
- A *module* for a chamfer rule could be developed from another pre-developed *module*, overhang.
- A *module* for a support structure rule could be developed from two other pre-developed *modules*, overhang and surface roughness.

Other examples that are common to each of those seven categories of AM processes, and thus process-independent, are also listed in Table 3.

Table 3. An example set of *modules* common to AM processes

No. of *module*	Definition
1	surface finish
2	overhang
3	feature allowable
4	shrinkage
5	tolerance
6	lattice structure
7	support structure
8	chamfer/fillet
9	rib enforcement

Using both sets of *modules* and *primitives*, we can introduce 'conditional sentences' into design rules as written above. Conditional sentences will allow design rules to be tailored for specific implementations while maintaining the independent functionalities of *modules*. This clearly indicates that standardizing their fundamental principles will support customization for specific AM processes, technologies, and applications. The following case study provides an example of explicit interpretations of design rules specific to a PBF process.

4 A CASE STUDY OF DESIGN RULE REDEFINITION WITH MODULARITY

An example set of design rules, introduced from a commercial catalogue of design guidelines specific to a PBF process [14], is shown in Fig. 2.

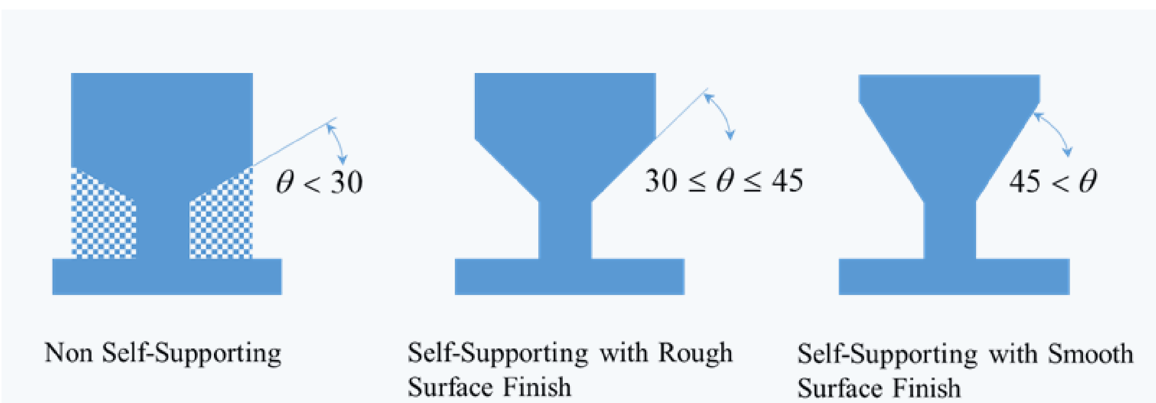

Fig. 2 An example set of commercial design guidelines introduced for the case study [14]

In this example, changes in the undercut angle of geometry impact the ability of the overhang to be self-supporting. If the undercut angle $\geq$ 45 degrees, the overhang will prove self-supporting. If the undercut angle falls in an interval (30 degrees to 45 degrees), then the overhang will only be self-supporting provided it has a rough surface finish. The subsidiary physical interrelations between the surface features (roughness) and the surrounding powder material makes self-supporting possible.

4.1 *Primitive* Definitions

An example set of *primitives* $\{P\}$ made for the case study is given in Table 4, where elements belonging to $\{P\}$, among others, are defined as process-independent over different AM processes. Short definitions of the *primitive* are also given in Table 4.

Table 4. An example set of *primitives* $\{P\}$ defined for the case study

Element symbol	Abbreviation	Definition
P_1	UC(undercut angle)	An angle created by offset layering build
P_2	SR(scan resolution)	A process resolution on x-y plane
P_3	RM(raw material type)	A characteristic of the material used
P_4	FD(feature dimension)	A dimensional scale of a feature during build
P_5	LT(layer thickness)	A layer thickness during build

4.2 *Module* Definitions

An example set of *modules* $\{M\}$ made for the case study is given in Table 5. Elements belonging to $\{M\}$, among others, are derived from $\{P\}$ and other pre-derived *modules* belonging to {M}, and *f*(x) is function of x.

Table 5. An example set of *modules* $\{M\}$ derived for the case study

Element symbol	Abbreviation	Function definition
M_1	OH(overhang)	$f(P_1, P_4)$
M_2	SF(surface finish)	$f(P_2, P_3\ P_5)$
M_3	SS(support structure)	$f(M_1, M_2)$

The process of generating an example set of *modules* $\{M\}$ is illustrated in Fig. 3. A *module* M_1 (OH: overhang) can be derived from two associated *primitives*, P_1 (UC: undercut angle) and P_4 (FD: feature dimension), each of which has a certain range of values. Another *module* M_2 (SF: surface finish) can be derived from three associated *primitives*, P_2 (SR: scan resolution), P_3 (RM: raw material type), and P_5 (LT: layer thickness), each of which has a certain range of values or types. Finally, the third *module* M_3 (SS: support structure) can be derived from M_1 and M_2 that are previously derived. Elements of $\{M\}$ are process-independent over different AM processes.

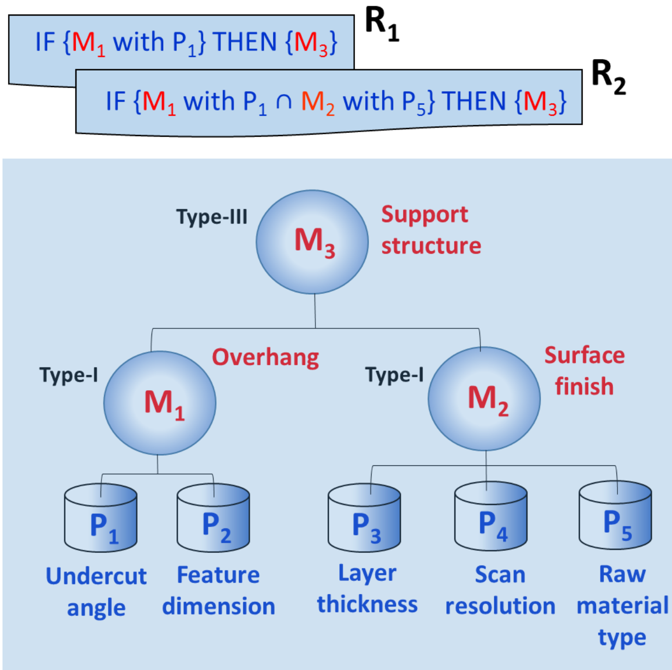

Fig. 3 A set of design rules $\{R\}$ generated using modular components $\{P\}$ and $\{M\}$ for the case study

4.3 Redefinition of Design Rules

An example set of design rules, $\{R\}$, redefined from the ones written above and shown in Fig. 2, is given in Table 6. Elements belonging to $\{R\}$ are derived from both $\{P\}$ and $\{R\}$, where relationships between the dependencies are outlined by the design principles.

Table 6. An example set of design rules $\{R\}$ compared with different representations

Element symbols	Type of representation	Design rule representation
R_1	Prescriptive guideline	*If given an* Overhang *with an* Undercut *of less than* 30 *degrees*, *then* Support Structures *are needed.*
	IF-THEN statement	IF $\{M_1$ with $P_1\}$ THEN $\{M_3\}$
	Description logic	$R_1 : M_1$ (OH1) $\bigcap$ LessThan (UC1,30) $\rightarrow M_3$ (SS1)
R_2	Prescriptive guideline	*If given an* Overhang *with an* Undercut *of* greater than 30 *degrees* and less than 45 *degrees and the* surface roughness *is* less than '1', *then* Support Structures *are needed.*
	IF-THEN statement	IF $\{M_1$ with $P_1 \bigcap M_2$ with $P_3\}$ THEN $\{M_3\}$
	Description logic	$R_2 : M_1$ (OH1) $\bigcap$ GreaterThan (UC1,30) $\bigcap$ LessThan (UC1,45) $\bigcap$ M_2 (RS1) $\bigcap$ LessThan (RS1,1) $\rightarrow M_3$ (SS1)

The consequents are 'functions of' the antecedents, and the exact relationship established will depend on the process and the intent of the rule.

5 CONCLUSIONS

This paper proposed a methodical approach to developing design rules with modularity for additive manufacturing. We demonstrated that the use of process-independent representations provide a basis for explicit interpretations of specific design rules, specifically for the powder-bed fusion process. Conclusions about design rules with modularity have been made as follows:

- First, they enable repeatability in an AM design guideline under different and diverse design and process environments.
- Second, they can communicate process-independent design principles to users that are unfamiliar with AM processes.
- Third, they are comprehensible and, therefore, manageable when subjected to changes and updates of AM machines and processes.

Modularity is applied to not only the geometric representations of the design, but also to the data providing support for the manufacture of the design. By defining design rules as sets of modular components, they can be more easily interpreted and implemented, independent of context. Since design rules impact process planning as well, we will also look to abstract recurring themes out of the process-specific guidelines.

Future research will continue investigating the use of our methodology on more complicated examples. We will demonstrate the reconfiguration of design rules from individual components, including process-specific parameters for different AM builds and processes. We will promote the consistent application of design rule principles as a means for mitigating ambiguities that may be introduced both within and between processes. We will continue to develop an extended design automation paradigm for the design and fabrication of new AM parts.

REFERENCES

[1] http://www.merriam-webster.com/dictionary/rule (accessed June 30, 2015).

[2] Gausemeier, J., Echterhoff, N., Kokoschka, M., Wall, M. (2011), *Thinking ahead the Future of Additive Manufacturing – Analysis of Promising Industries*, available at: http://dmrc.uni-paderborn.de/fileadmin/dmrc/Download/data/DMRC_Studien/DMRC_ Study.pdf (accessed June 30, 2015).

[3] Suzuki, K. (1998), *Method of checking design rules for semiconductor integrated circuit*, United States Patent 5706295, 01/06/1998.

[4] http://en.wikipedia.org/wiki/Design_rule_checking (accessed June 25, 2015).

[5] ASTM-F42.91 Subcommittee (2015), *Standard terminology for additive manufacturing technologies*, ASTM Designation: F2792-12a, Vol.10.04.

[6] Kranz, J., Herzog, D., and Emmelmann, C. (2015), Design guidelines for laser additive manufacturing of lightweight structures in TiAl6V4, *J. of Laser Applications*, 27, S14001.

[7] Stratasys Co., *TECHNICAL APPLICATION GUIDE: FDM for End-Use Parts: Tips and Techniques for Optimization*, available at: http://usglobalimages.stratasys.com/Main/Secure/Applications/End%20Use%20Parts/SSYS-TAG-End-UseParts-11-13.pdf (accessed June 25, 2015).

[8] RedEye Co., *Design for Additive Manufacturability:FDM Basics*, available at: https://www.stratasysdirect.com/wp-content/uploads/2015/07/fdm-basics.pdf (accessed June 25, 2015).

[9] ProtoPafadigm Co., Printing With Support - Extreme Overhangs, available at: http://www.protoparadigm.com/news-updates/printing-with-support-extreme-overhangs (accessed June 30, 2015).

[10] Thomas, D. (2009), *The Development of Design Rules for Selective Laser Melting*, Ph. D. Thesis at U of Wales Institute.

[11] EOS Co., *Additive Manufacturing (AM) Basic Design Rules for Additive Manufacturing, eos, e-manufacturing solution*, available at: https://ip-saas-eos-cms.s3-eu-west-1.amazonaws.com/public/ab4f0542d66453fc/5f889ab7e3f72bd3d44b22205ba8b68b/EOS-Basic-Design-Rules_Additive-Manufacturing_EN.pdf (accessed June 30, 2015).

[12] Wang, D., Yang, Y., Liu, R., Xiao, D. and Sun, J., (2013), Study on the designing rules and processability of porous structure based on selective laser melting (SLM), *Journal of Materials Processing Technology, Journal of Materials Processing Technology*, 213, 1734–1742.

[13] Samperi, M., (2014), *Development of Design Guidelines for Metal Additive Manufacturing and Process Selection*, M.Sc Thesis, The Penn State University.

[14] Xometry Co., *DMLS Design Guide*, available at: http://cdn2.hubspot.net/hub/340051/file-2443491006-pdf/Design_Guides/NextLine_DMLS_Design_Guide.pdf?submissionGuid=0fbbb592-3eb6-4b3a-a083-439f2e35b22e (accessed June 30, 2015).

[15] Seepersad, C.C, Govett, T., Kim, K., Lundin, M., and Pinero, D. (2012), A designer's guide for dimensioning and tolerancing SLS parts, *Proceedings of the Solid Freeform Fabrication International Symposium*, University of Texas, Austin, Texas, 921-931.

[16] Adam, A.O. G. and Zimmera, D. (2014), Design for Additive Manufacturing—Element transitions and aggregated structures, *CIRP Journal of Manufacturing Science and Technology*, 7 (1) 20–28.

[17] Gibson, I., Goenka, G., Narasimhan, R. and Bhat, N. (2010), Design rules for additive manufacture, *Proceedings of the Solid Freeform Fabrication International Symposium*, University of Texas, Austin, Texas, 705-716.

[18] Geometric Co., (2013), *Design for Additive Manufacturing Version 1.0*, available at: http://geometricglobal.com/wp-content/uploads/2013/11/Geometric_Whitepaper-Additive-Manufacturing_July-2013.pdf.pdf (accessed June 25, 2015).

[19] Hietikko, E., (2014), Design for Additive Manufacturing – DFAM, *The International Journal Of Engineering And Science (IJES)*, 3 (12) 14-19.

[20] Filippi, S. and Cristofolini, L., (2009), *The Design Guidelines Collaborative Framework: A Design for Multi-X Method for Product Development*, ISBN 1848827725, 9781848827721, Springer Science & Business Media.

[21] Zhang, Y., Bernard, A., Harik, R. and Karunakaran, K.P. (2015), Build orientation optimization for multi-part production in additive manufacturing, *Journal of Intelligent Manufacturing*, 10.1007/s10845-015-1057-1.

[22] Jee, H. and Suh, J., (2013), Software supports and E-manufacturing for DMT process, *J. of Manufacturing Science and Tech*., 27 (10) 2947-2953.

[23] Bessiere, C., (2006), *Constraint propagation*, F. Rossi, P. van Beek, T. Walsh (Eds.), Handbook of Constraint Programming, Elsevier (Chapter 3).

[24] Haspelmath, M., König, E., Oesterreicher, W. and Raible, W., (2001), *Language Typology and Language Universals*, Walter de Gruyter.
[25] Thane, C., (2010), *Teacher Training Essentials: Workshops for Professional Development*, Cambridge University Press.
[26] http://www.oxforddictionaries.com/us/definition/american_english/module (accessed June 30, 2015).