

# Regelbasierte Analyse von Entscheidungen im Entwurf komplexer Systeme

Johannes Groß<sup>1</sup>, Stephan Rudolph<sup>1</sup>

<sup>1</sup>Universität Stuttgart, Institut für Statik und Dynamik der Luft- und Raumfahrtkonstruktionen, Pfaffenwaldring 27, 70569 Stuttgart,  
[gross@isd.uni-stuttgart.de](mailto:gross@isd.uni-stuttgart.de), [rudolph@isd.uni-stuttgart.de](mailto:rudolph@isd.uni-stuttgart.de)

**Zusammenfassung:** Im Entwurf komplexer Systeme werden meist viele der zentralen Entscheidungen ohne umfassende Kenntnis aller damit zusammenhängenden Folgen getroffen. Eine der Ursachen hierfür ist, dass diese Kenntnis eine manuell sehr aufwändige Ausarbeitung der damit verbundenen Entwurfsalternativen erfordert. Dagegen können in einer graphenbasierten Entwurfssprache durch den grundsätzlich regelbasierten Aufbau des Systems die Entwurfsschritte und Entscheidungen automatisiert ausgeführt werden. Im Bedarfsfall können die einzelnen Entwurfsschritte damit immer wieder von neuem ausgeführt werden. So ist es möglich, in der Explorationsphase partiell Entscheidungen zurückzunehmen (sog. Backtracking). Dadurch kann in einer bestimmten Entscheidungssituation durch regelbasierte Suche in dem Entwurfsraum das erforderliche bessere qualitative Verständnis und quantifizierte Wissen über die Auswirkungen einzelner Entwurfsentscheidungen auf das Gesamtsystem erzeugt werden.

## 1 Entwurf komplexer Systeme

Der Entwurf komplexer Systeme ist durch eine Vielzahl starker wechselseitiger Abhängigkeiten der Entwurfsparameter gekennzeichnet. Um trotzdem verschiedene Funktionen entsprechend den Anforderungen erfüllen zu können, ist es notwendig, aus einer Menge von Alternativen die best geeigneten Lösungsprinzipien auszuwählen und auf vorteilhafte Weise miteinander zu koppeln. Denn diese ausgewählte Kombination der Bauteile, die letztendlich nach diesen Lösungsprinzipien funktionieren, ergibt miteinander das komplexe (Gesamt-)System. Jedoch können durch die wechselseitigen Abhängigkeiten der Bauteile untereinander und den daraus zusammengesetzten Subsystemen auf der höheren Systemebene auch verschiedene, im Voraus nicht unbedingt vorhersehbare Effekte auftreten. Daher ist es wichtig, die Entscheidungslogik, mit welcher nacheinander an den verschiedenen Stellen im Entwurfsprozeß und auf den verschiedenen hierarchischen Ebenen (d.h. der Bauteil-, Subsystem- und Gesamtsystemebene) für oder gegen den Einsatz bestimmter Lösungsprinzipien und/oder Bauteile entschieden wird, genauer zu analysieren.

In dieser Darstellung wird anhand eines konkreten Beispiels aufgezeigt, wie sich ein mehrstufiges Entscheidungsschema auf die Abdeckung des Entwurfsraumes in einem Subsystem auswirkt. Die Umsetzung des Entscheidungsschemas erfolgt hierbei mit einer graphenbasierten Entwurfssprache [Alb03] auf Basis der UML (Unified Modeling Language). Die Verarbeitung der Entwurfssprache erfolgt mit dem Entwurfscompiler 43v2 [IILS]. Die Wahl der Modellierungssprache ist durch den Entwurfscompiler 43v2

historisch bedingt. Die UML war zum Zeitpunkt der Entwicklung der Methodik bereits zu einer international anerkannten Norm gereift und es waren Implementierungen in verschiedenen Open Source Tools verfügbar. Aus heutiger Sicht könnte hierfür auch die speziell für das Systems Engineering aus der UML abgeleitete Version der SysML (Systems Modeling Language) für die Definition von Entwurfssprachen in Betracht kommen, es wären jedoch einige zusätzliche Anpassungen durchzuführen.

## 2 Beispielsystem Energieversorgung

In dieser Darstellung soll für die fiktive Satellitenmission „FireSat“ das Energieversorgungssystem ausgelegt werden. Die Mission „FireSat“ ist aus einem Entwurfshandbuch für Raumfahrtssysteme entnommen [LaW99]. Es handelt sich dabei um einen Satellit für das Auffinden und Überwachen von Waldbränden in den USA. Das beschriebene Teilsystem für die Energieversorgung ist in eine umfassende Entwurfssprache für das Gesamtsystem eingebunden. In [Gro11] wird die durch die Kopplung der Bauteile zu Subsystemen entstehende Entscheidungshierarchie erstmals vorgestellt. In [Gro12a] wird ein Überblick über den regelbasierten Aufbau des Gesamtsystems gegeben und die Generierung von Analysemodellen aus der Entwurfssprache vorgestellt. In [Gro12b] werden dazu weitere Aspekte der Abhängigkeitsanalyse in einem Satellitensystem vorgestellt. In dieser Darstellung stehen der Aspekt des regelbasierten Aufbaus eines Subsystems und die daraus entstehende Möglichkeit zur Untersuchung des Entwurfsraums und der darin getroffenen Entscheidungen im Vordergrund.

### 2.1 Modellierung des Energieversorgungssystems (in UML)

Das Energieerzeugungssystem besteht auf oberster (System-)Ebene aus fünf Klassen. Diese sind in Bild 1 dargestellt. Die Klasse „PowerSystem“ repräsentiert das (Gesamt-)

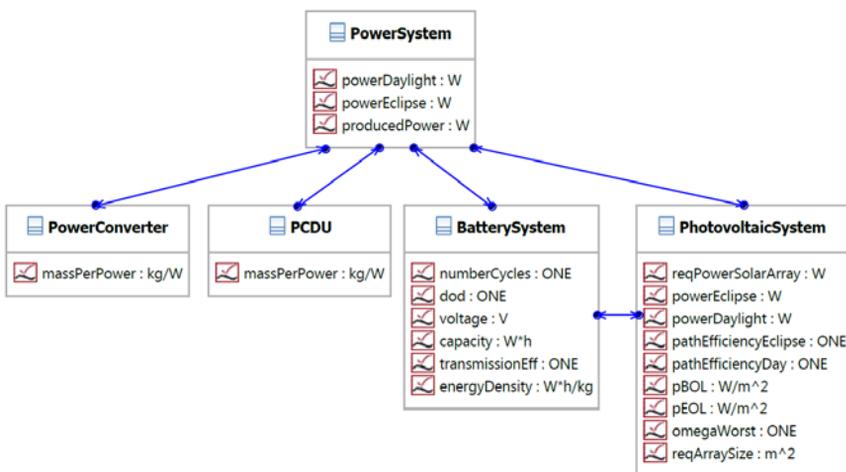


Bild 1: Klassen des Energieversorgungssystems

System. Dort wird der Bedarf an Energie im Tagesmodus sowie während der Schattenphase („powerEclipse“) vorgegeben. Unterhalb der Gesamtsystemklasse „PowerSystem“ sind die verschiedenen Einzelbestandteile des Systems aufgeführt. Das System setzt sich aus einer Energiewandlereinheit („PowerConverter“), der Stromverteilungseinheit („Power Control and Distribution Unit – PCDU“), einem Batteriesystem („Battery-System“) sowie dem Stromerzeugungssystem („PhotovoltaicSystem“) zusammen. Für die Stromerzeugung wird in diesem Fall ein photovoltaisches System umgesetzt.

Das Photovoltaiksystem selbst wiederum gliedert sich in die in Bild 2 dargestellten weiteren Klassen auf. Es gibt zum einen die Solarzellen („SolarCell“), die eine unterschiedliche Effizienz aufweisen (können). Die drei verschiedenen Solarzellen sind in diesem Beispiel farbig markiert. Das hellste grün steht für die kostengünstigsten Zellen mit der geringsten Effizienz. Dunkler werdend steigen die Effizienz der Zellen und somit auch die Kosten dafür an. Auf der anderen Seite gibt es die verschiedenen Arten von Paneelen des Photovoltaiksystems. Der Satellit kann ein direkt auf die Struktur montiertes Panel haben („BodyPanel“). Neben dem direkt montierten Panel können zwei weitere klappbare Paneele („FlapPanel“) montiert werden. Sollten diese nicht ausreichen um den Leistungsbedarf zu befriedigen, so können nochmals außen zwei weitere Paneele („ExtensionPanels“) montiert werden.

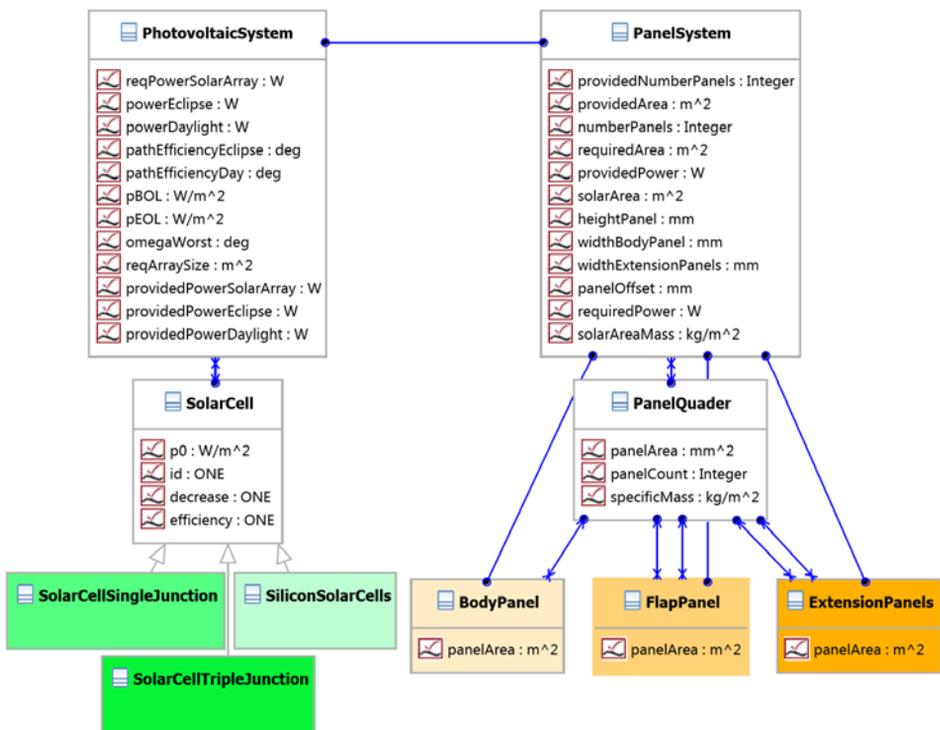


Bild 2: Klassen des Photovoltaik-Systems

In den Panels in Bild 2 sind die Gleichungen zur Berechnung der Fläche enthalten. Aus der Fläche wird u.a. auch in Verbindung mit dem Wert „Power End of Life – pEOL“ der Klasse „PhotovoltaicSystem“ die Leistung eines Panels berechnet. Dieser Wert wird über alle Paneele aufsummiert und damit die Leistung des Systems berechnet. Dem wird der Bedarf des Gesamtsystems an Energie gegenübergestellt, der aus dem Bedarf in der Licht- sowie in der Schattenphase bestimmt wird. In der Folge ist das Energiesystem so auszulegen, dass der Energiebedarf über einen Orbit hinweg gestillt werden kann.

## 2.2 Regeln für den Aufbau des Systems

Die oben vorgestellten Klassen werden für eine Ausprägung des Systems durch UML *InstanceSpecifications* definiert. In diesen Instanzen sind die Werte für die in den Klassen vorgegebenen Attribute abgespeichert. Die Instanzen werden durch graphische Regeln erstellt. In den graphischen Regeln wird auf der linken Seite eine Suche auf dem vorhandenen Modell definiert und auf der rechten Seite eine Transformation des vorhandenen Modells festgelegt. In Bild 3 ist die erste Regel für den Aufbau des Solarpanelsystems abgebildet.

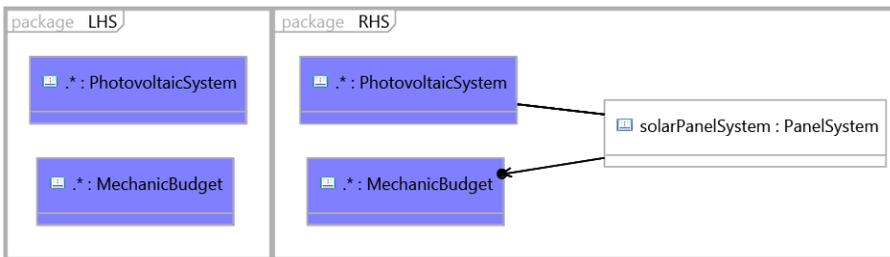


Bild 3: Regel für den Aufbau des Solarpanelsystems

Auf der linken Seite der Regel in Bild 3 werden zwei Instanzen gesucht. Eine dieser Instanzen muss von der Klasse "PhotovoltaikSystem" sein und die andere von der Klasse "MechanicBudget". Dies ist eine Klasse aus einem übergeordneten Modell der Satellitenentwurfssprache. Die beiden Instanzen werden mit einer neu erstellten Instanz der Klasse "PanelSystem" verbunden.

In Bild 4 ist ein so genannter Cast dargestellt. Dies bezeichnet den Wechsel der Klasse einer Instanz. In dieser Regel wird eine Instanz mit der Klasse "SolarCellSingleJunction"



Bild 4: Regel zum Wechsel (Casting) der Klasse einer Instanz

gesucht und dann in eine Instanz der Klasse "SolarCellTripleJunction" umgewandelt (bzw. ersetzt). Dies bewirkt, dass die Gleichungen der vorherigen Klasse nicht mehr auf dieser Instanz ausgeführt werden, dafür werden die Gleichungen der Klasse "SolarCellTripleJunction" auf der Instanz ausgeführt.

In Bild 5 ist die Regel zum Einbau des "BodyPanel" dargestellt. Auf der linken Seite der Regel wird das "PanelSystem" gesucht und auf der rechten Seite eine Instanz des "BodyPanel" sowie eine Instanz des "PanelQuader" hinzugefügt. Die Instanz "panelMQuader" beinhaltet dazu die geometrische Repräsentation des Solarpanels.

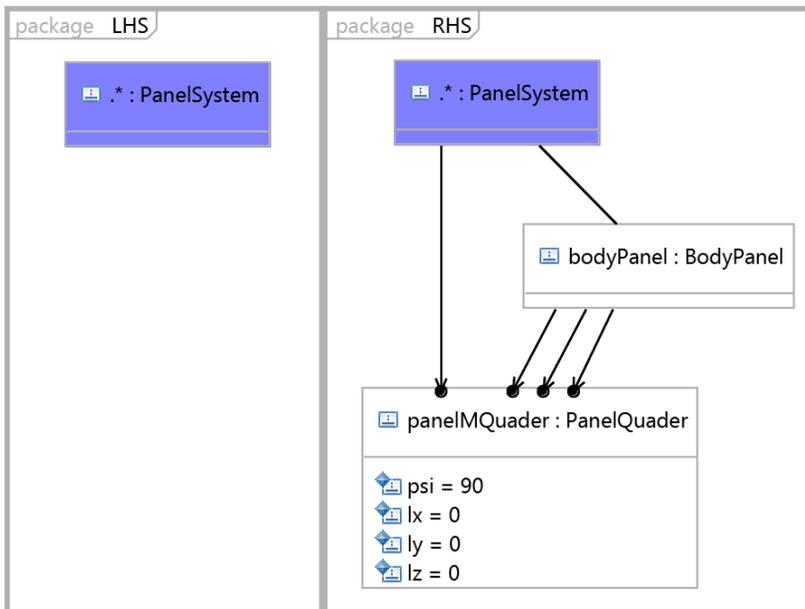


Bild 5: Regel zur Erstellung des mittleren Solarpanels

## 2.3 Modellierung der Entwurfslogik

Neben den bereits vorgestellten Regeln gibt es noch einige weitere, meist ähnliche Regeln um die verschiedenen Varianten des Energieversorgungssystems ausprägen zu können. Der Aufbau des Systems folgt einer ebenfalls graphisch programmierten Entwurfslogik in Form eines so genannten Aktivitätsdiagramms. Um den Bedarf an Energie mit möglichst geringem Aufwand zu erfüllen, wurde folgende Logik vorausgesetzt:

- Effizientere Solarzellen sind teurer als weniger effiziente
- Ein Panel mehr ist teurer als bessere Zellen einzusetzen.

In Bild 6 ist ein ausführbares Aktivitätsdiagramm zum Aufbau des Systems dargestellt.

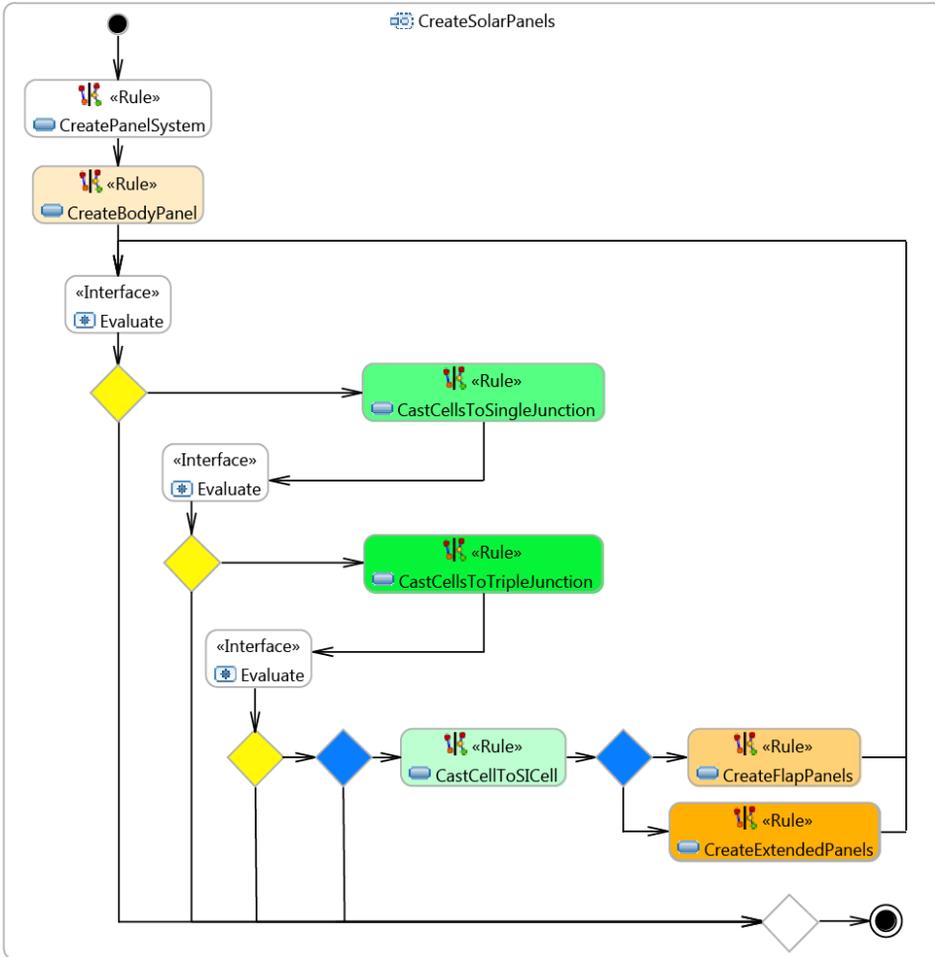


Bild 6. Aktivitätsdiagramm zum Aufbau des Energieversorgungssystems

Zunächst wird im Aktivitätsdiagramm in Bild 6 die Klasse „PanelSystem“ instanziiert. Dann wird das auf dem Satelliten montierte Panel mit der in Bild 5 dargestellten Regel angebracht. In den gelb markierten Entscheidungsknoten (UML *DecisionNodes*) wird dann nach der Berechnung ("Interface Evaluate") überprüft, ob das System mehr Energie liefert als mindestens notwendig. Wenn dies der Fall ist, endet der Programmablauf. Ist dies nicht der Fall, so folgt der Programmablauf dem horizontalen Pfeil, um die Leistungsfähigkeit des Systems durch den Wechsel zu besseren Zellen zu steigern. Der Wechsel wird durch einen Cast der Instanz der Oberklasse "SolarCell" erreicht, wie dies in Bild 4 gezeigt wurde. Sollte der Wechsel zu besseren Zellen nach den ersten beiden Entscheidungsknoten nicht genügen um die Anforderungen zu erfüllen, so werden nach dem dritten gelben Entscheidungsknoten wieder die „billigsten“ Zellen eingebaut.

Danach werden mit der Regel "CreateFlapPanels" zwei weitere Panele an den Satelliten angebaut. Mit diesen weiteren Panele wird der Zyklus für die Auswahl der Zellen erneut durchlaufen. In dem blauen Entscheidungsknoten vor den beiden Regeln zum Einbau der Panels wird ein Zähler beim Durchlaufen des Knotens inkrementiert. Dadurch wird der Fluss beim zweiten Durchlaufen des Knotens auf den anderen Ausgang gelenkt und damit die Regel "CreateExtendedPanels" ausgeführt. Der blaue Knoten der nach dem gelben Knoten kommt, beendet bei einem dritten Durchlauf das Aktivitätsdiagramm da das Energiesystem in diesem Fall die Forderungen nicht erfüllen kann. In diesem Fall liegt kein valider Satellitendesign vor.

Sollte die Entscheidungslogik in einem Aktivitätsdiagramm auf mehreren Parametern beruhen, so sind diese in dem formalen Ansatz der Entwurfssprachen zueinander in Beziehung zu setzen. Wenn man das Beispiel einer zweidimensionalen Paretofront betrachtet, so sind dort zwei Bewertungsgrößen übereinander aufgetragen. Diese gehen scheinbar unabhängig in die Entwurfsentscheidung ein. Legt man nun jedoch eine Steigung der Front fest, um den Optimumspunkt an der Paretofront zu finden, so werden die beiden Bewertungsgrößen für den Entwurf durch diese Steigung gekoppelt. Dadurch lässt sich die Entscheidung wieder auf einen Parameter reduzieren. Allgemein kann man daraus schließen, dass es nur in unklaren oder nicht ausreichend formalisierten Entwurfsgebieten „wirkliche“ Entscheidungen gibt, die auf mehreren Parametern beruhen, da es nicht gelingt, eine für die konkrete Situation gültige Aggregationsfunktion zu finden bzw. festzulegen.

## 2.4 Ausführung der Entwurfssprache

Wird diese Aufbau-logik nun mit den Randbedingungen eines gegebenen Satellitendesigns ausgeführt, so kann man durch die Variation der Anforderungen an das Energiesystem den Entwurfsraum des Subsystems untersuchen. In Bild 7 ist dieser Entwurfsraum über zwei Achsen dargestellt. Horizontal ist die geforderte Leistung aufgetragen, vertikale die vom Subsystem gelieferte Leistung. Die Funktion der gelieferten Leistung über dem Leistungsbedarf verläuft stufenförmig. Oberhalb der roten Winkelhalbierenden ist der Entwurf valide da mehr Energie produziert wird als gefordert ist. Unterhalb der roten Linie liefert das Energieversorgungssystem nicht genügend Leistung und der Entwurf ist ungültig.

Wie aus der Farbabstufung in Bild 7 gefolgert werden kann, ist ab einer geforderten Leistung von 0 bis rund 60 Watt (blaue Linie) das Energieversorgungssystem im Entwurf zunächst auf unterster Ebene mit der einfachsten Konfiguration (BodyPanel und Silikon-Zellen) ausgelegt. Wenn die Leistungsanforderungen diesen Wert übersteigen, werden bessere Zellen eingebaut. Dies verursacht einen Schritt nach oben in der Treppe. Wenn diese Leistung nicht mehr ausreicht, werden wieder bessere Zellen eingebaut. Dies ist durch die zweite Stufe repräsentiert. Wenn die Leistung nun nicht mehr ausreicht, werden zwei Panels zusätzlich am Satellit angefügt und die Zellen wieder auf die günstigste Variante zurückgestellt. Danach beginnt der Kreislauf von vorne.

### Gelieferte Leistung [W]

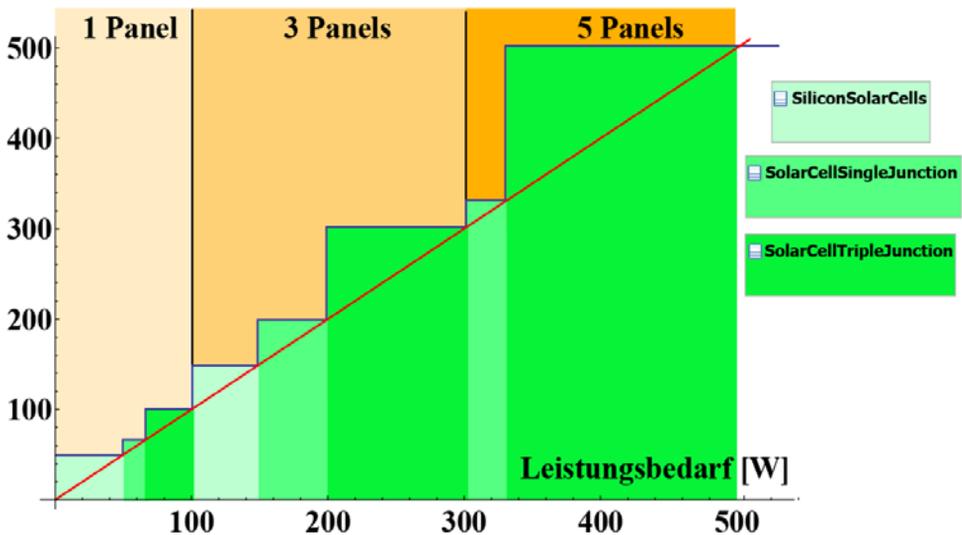


Bild 7: Leistungsangebot des Systems über der geforderten Leistung

Wie man sieht werden die Stufen zwischen den Schritten, in welchen die Zellen verbessert werden, bei höheren Leistungen immer größer. Dies liegt daran, dass eine größere Panelfläche die Effizienzsteigerung der Zellen stärker zur Geltung kommen lässt. Die Stufe bei einem Leistungsbedarf von 300W von einem System mit 3 Panels und den besten Zellen zu einem System mit 5 Panels und den mittleren Zellen ist hingegen sehr klein. Dies liegt daran, dass bei größerer Panelfläche der Zugewinn durch ein zusätzliches Panelpaar geringer wird im Verhältnis zu dem Zugewinn durch bessere Zellen. Dies führt dazu, dass die Topologie der schlechtesten Solarzellen mit 5 Panels an dem Satellit bei dieser Modellierung nicht mehr auftaucht. Es lässt sich daraus also für dieses Modell in diesem Kontext schlussfolgern, dass je größer das Photovoltaiksystem ist, desto bessere Zellen sollten eingesetzt werden.

### 3 Zusammenfassung

Es wurde gezeigt, wie mit einfachen Mitteln die Auswirkungen von Entscheidungen auf den unterschiedlichen Hierarchieebenen von Bauteil-, Subsystem- und Gesamtsystementwurf maschinell untersucht werden können. Das regelbasierte Vorgehen dient in diesem Fall ähnlich einem Schachprogramm mit Bewertungsfunktion der Tiefensuche im Entwurfsraum. Das Schachprogramm kennt die Regeln des Spiels und kann damit gültige Züge auf dem Spielbaum ausführen. Eine Entwurfssprache kennt analog durch die Regeln gültige Veränderungen des Entwurfs und kann damit in Entscheidungssituationen die Auswirkungen durch eine Suche auf dem Entwurfsbaum genauer (d.h. durch eine weitere Detaillierung des Entwurfs) untersuchen.

Das hier vorgestellte Subsystem für die Energieversorgung ist im Rahmen der Entwurfsaufgabe „FireSat“ in eine weitaus umfassendere Entwurfssprache für das Gesamtsystem Satellit eingebettet. Die Beschreibung dieser Entwurfssprache ist in der Anfang nächsten Jahres erscheinenden Dissertation des Erstautors enthalten. Dort werden alle auf Bauteil-, Subsystem- und Systemebene durchgeführten Untersuchungen im Detail vorgestellt. Der Nachweis der Modellierung eines komplexen Gesamtsystems mit den hier vorgestellten Mitteln ist somit kurz vor seiner Vollendung. Der Mehrwert im Entwurf und der Analyse eines komplexen Systems liegt dabei in der Verkürzung der Entwurfszeit, der Verbesserung der Ergebnisse durch mögliche Optimierungen, der Evaluierungsmöglichkeit verschiedener Topologien und der Wiederverwendbarkeit bereits vorhandener Entwürfe und des in den Klassen und Regeln abgespeicherten Entwurfswissens.

## Literaturverzeichnis

- [Alb03] R. Alber, S. Rudolph “43” – a generic approach for engineering design grammars  
Proceedings AAAI Spring Symposium ‘Computational Synthesis’, Stanford (2003)
- [IILS] Designcompiler 43, IILS GmbH, <http://www.iils.de>.
- [Gro12a] Groß, J. Rudolph, R.: Generating Simulation Models from UML – A FireSat Example.  
In Proc. of the *Symposium on Theory of Modeling and Simulation (TMS’12)*. Orlando, FL (USA) 26-29 March 2012.
- [Gro12b] Groß, J., Rudolph, R.: Dependency Analysis in Complex System Design using the FireSat Example. Proceedings of the INCOSE Symposium, Rome (Italy), 2012.
- [Gro11] Groß, J., Rudolph, R.: Hierarchie von Entwurfsentscheidungen im modellbasierten Entwurf komplexer Systeme. Tag des Systems Engineering der GfSE, Hamburg (Germany) 2011.
- [LaW99] Larson, W. J., and J. R. Wertz, eds.: Space mission analysis and design. 3rd ed. El Segundo, California: Microcosm, 1999.