

METHODS AND SYSTEMS FOR AUTOMATED SPACECRAFT DESIGN**CROSS-REFERENCE TO RELATED APPLICATION**

[0001] This application claims the benefit of and priority to U.S. Provisional Patent Application No. 63/626,932 filed January 30, 2024, U.S. Provisional Patent Application No. 63/626,946 filed January 30, 2024, and U.S. Provisional Patent Application No. 63/564,095 filed March 12, 2024, the entire disclosure of each of which is hereby incorporated by reference in its entirety for all purposes.

TECHNICAL FIELD

[0002] The present disclosure relates generally to the design and delivery of spacecrafts, and more specifically, to software-based methods and systems for rapid design and delivery of spacecrafts.

BACKGROUND

[0003] Satellite structural design is typically generated by an engineer at a computer using Computer Aided Design (CAD) software and then exported to another program for detailed structural analysis such as Finite Element Analysis (FEA) or sometimes to a different application in the same CAD toolset. These operations, design, and detailed structural analysis are quite labor intensive and require highly-skilled subject matter experts, and thus the efficiency of the current satellite structural design is quite low. For example, the design and delivery of an Evolved Expendable Launch Vehicle (EELV) Secondary Payload Adapter (ESPA)-class satellite generally takes between 28 and 48 months (including 14-30 months for mission formulation and satellite design).

[0004] The foregoing examples of the related art and limitations therewith are intended to be illustrative and not exclusive, and are not admitted to be “prior art.” Other limitations of the related art will become apparent to those of skill in the art upon a reading of the specification and a study of the drawings.

SUMMARY

[0005] According to some embodiments, the present disclosure provides a method for automated spacecraft design. The method includes receiving mission parameters provided by a client for spacecraft design, generating one or more concept designs based on the received mission parameters, a concept design including one or more components selected for each subsystem included in a designed spacecraft, for the concept design, automating a generation of a frame design that accommodates the one or more components selected for each subsystem included in the designed spacecraft, the frame design being a CAD model that includes individual frame pieces generated, sized and fitted automatically via automated scripts compatible with a CAD system, including one or more structural design software applications; performing a series of analyses or simulations to evaluate a performance of the frame design; and generating a digital twin representing a virtual representation of the designed spacecraft based on the generated frame design and the series of analyses and simulations.

[0006] The foregoing is a summary and thus contains, by necessity, simplifications, generalizations and omissions of detail; consequently, the summary is illustrative only and is not limiting in any way. Other aspects, inventive features, and advantages of the systems and/or processes described herein will become apparent in the non-limiting detailed description set forth herein.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The accompanying figures, which are included as part of the present specification, illustrate the presently preferred embodiments and together with the general description given above and the detailed description of the preferred embodiments given below serve to explain and teach the principles described herein.

[0008] FIG. 1 is a flowchart of a process for designing and delivering a spacecraft, in accordance with some embodiments.

[0009] FIG. 2A is a flowchart of a conceptual design method of a spacecraft design process, in accordance with some embodiments.

[0010] FIG. 2B is an illustration of input/output of a conceptual design method of a spacecraft design process, according to an example.

- [0011] FIG. 3A is a flowchart of a preliminary design method of a spacecraft design process, in accordance with some embodiments.
- [0012] FIG. 3B is an illustration of input/output of a preliminary design method of a spacecraft design process, according to an example.
- [0013] FIG. 3C illustrates example outputs at different stages of a spacecraft design process, according to one example.
- [0014] FIG. 4 illustrates a process flow of a conceptual design method of a spacecraft design process, in accordance with some embodiments.
- [0015] FIGS. 5A-5D illustrate various subsystem selections in a spacecraft design process, in accordance with some embodiments.
- [0016] FIG. 6 illustrates a system architecture for automated spacecraft design, in accordance with some embodiments.
- [0017] FIG. 7 illustrates an example block diagram of a concept design of a spacecraft, in accordance with some embodiments.
- [0018] FIG. 8 illustrates an example outcome of a generative spacecraft design for additive manufacturing, in accordance with some embodiments.
- [0019] FIG. 9 is a flowchart of an example method for tradespace exploration in spacecraft design, in accordance with some embodiments.
- [0020] FIG. 10 is a block diagram of implementing a tradespace exploration in the spacecraft design, in accordance with some embodiments.
- [0021] FIG. 11 is a block diagram of an example computer system, in accordance with some embodiments.
- [0022] It will be appreciated that, for simplicity and clarity of illustration, where considered appropriate, reference numerals may be repeated among the figures to indicate corresponding or analogous elements.

DETAILED DESCRIPTION

[0023] The present disclosure describes software-based methods and systems for rapid design and delivery of spacecrafts (such as space vehicles and satellites among others). The Figures (FIGS.) and the following description relate to preferred embodiments by way of

illustration only. It should be noted that from the following discussion, alternative embodiments of the structures and methods disclosed herein will be readily recognized as viable alternatives that may be employed without departing from the principles of the disclosure.

A Process for Designing and Delivering a Spacecraft

[0024] FIG. 1 illustrates a process 100 for designing and delivering a spacecraft. In the example of FIG. 1, the design and delivery process includes four phases. Phase A (mission formulation) includes steps 102-103. Phase B (preliminary design) includes steps 104-105. Phase C (final design and design fabrication) includes steps 106-114. Phase D (assembly, integration, and testing) includes steps 116-121. It should be noted that these phases are traditionally defined by the NASA systems engineering handbook. Because the execution of the system disclosed herein is automated, the outlined steps may roughly correspond to the NASA guidelines.

[0025] Phase A (mission formulation) includes steps 102-103. In existing implementations, Phase A generally takes between 3 and 6 months.

[0026] In step 102, the spacecraft's mission is formulated. Mission formulation may involve defining the mission (often with input from the entity that will own or operate the spacecraft) and determining system requirements for the spacecraft based on the mission. In existing implementations of the process 100, step 102 is generally a manual, iterative process developed in the 20th century that is prone to scope creep with little to no digitization of requirements. In addition, at step 102, large part counts introduce substantial room for error and increased complexity in supply chain tracking, inspection, and assembly.

[0027] In step 103, a System Requirements Review (SRR) is conducted. The SRR examines the functional and performance requirements defined for the system to determine whether the requirements satisfy the mission.

[0028] Phase B (preliminary design) includes steps 104-105. In existing implementations, Phase B generally takes between 3 and 8 months.

[0029] In step 104, a preliminary design for the spacecraft is generated. The preliminary design step may involve using systems engineering techniques to develop a block diagram of the spacecraft system. Existing implementations of step 104 tend to be highly bespoke, such that the preliminary design is generated from scratch according to the unique requirements of

the mission. In existing implementations, the highly customized nature of the preliminary design step tends to be incompatible with modern, rapidly advanced manufacturing processes. Likewise, the highly customized approach to preliminary design tends to produce parts and subsystems with highly complex geometries that require significant design time to be manufacturable. Furthermore, in existing implementations, the structural and thermal design components of the preliminary design step tend to be manual processes reliant on individual know-how and often require Subject Matter Experts (SMEs).

[0030] In step 105, a Preliminary Design Review (PDR) is conducted. The purposes of the PDR may include (1) determining whether the preliminary design meets all system requirements with acceptable risk and within the cost and schedule constraints and (2) establishing the basis for proceeding with the final design.

[0031] Phase C includes two sub-phases, final design (steps 106-109) and design fabrication (steps 110-114). In existing implementations, the final design sub-phase generally takes between 8 and 16 months, and the design fabrication sub-phase generally takes between 8 and 10 months.

[0032] In step 106, subsystems of the spacecraft are procured. In step 108, spacecraft design and analysis are performed. In step 109, a Critical Design Review (CDR) is conducted. The purposes of the CDR may include (1) determining whether the final design meets all mission performance requirements within the identified cost and schedule constraints, and (2) establishing the basis for proceeding with full-scale design fabrication, assembly, integration, and testing.

[0033] In step 110, a flat spacecraft (FlatSat) is assembled. The FlatSat may include a motherboard where spacecraft avionics modules can be installed and connected as if inside the actual spacecraft structure, and may function as a ground-based testbed for the spacecraft (e.g., for CubeSat testing). In step 112, functional testing and software testing may be conducted. In step 114, the SV structure may be fabricated.

[0034] In existing implementations of Phase C, numerous practices give rise to substantial delay. For example, legacy Computer Numerical Control (CNC)--based processes can impose time-consuming and capability-limiting constraints on the design. So-called "standard buses" may require time-consuming modifications or may not be compatible with the mission requirements. A non-optimal theory of forcing new, sometimes revolutionary capabilities to

fit standardized designs can impose additional capability-limiting constraints. The absence of systematic operations process flow can cause additional delays.

[0035] Phase D (assembly, integration, and testing) includes steps 116-121. In existing implementations, Phase D generally takes between 6 and 8 months.

[0036] In step 116, the spacecraft subsystems and structure are assembled. In step 118, system functional testing is conducted. In step 119, a Test Readiness Review (TRR) is conducted. The purposes of the TRR may include determining whether the test system (hardware and software), test facility, support personnel, and test procedures are ready for environmental. In step 120, environmental testing is conducted. In step 121, a Program Status Review (PSR) is conducted.

[0037] In existing implementations of Phase D, numerous practices give rise to additional delay. For example, if there has been inadequate or minimal unit-level testing in the earlier Phases, the systems-level testing in Phase D may reveal many unit-level errors at a mature level of integration, resulting in significant delays. In addition, the testing process may require significant oversight by subject matter experts, because the testing protocols may not be rigorously defined and/or may involve only minimal automation.

Motivation for and Benefits of Some Embodiments

[0038] In existing implementations, the process 100 for designing and delivering a spacecraft (e.g., an ESPA-class or smaller spacecraft), including Phases A-D, generally takes between 28 and 48 months. Phases A, B, and the final design sub-phase of Phase C generally take between 14 and 30 months. Techniques for more rapid and less expensive design and delivery of spacecrafts are needed.

[0039] The present disclosure describes software-based methods and systems for rapid and inexpensive design and delivery of spacecrafts. In some cases, computer systems and/or software configured to perform the methods described herein may be referred to as “Spec to Spacecraft” (STS) systems (which may have a name of “MERCURY” system or another different name) and/or software. An STS system may include a mechanical engineering module (e.g., software module) configured to perform mechanical engineering tasks, an orbital modeling module (e.g., software module) configured to perform orbital modeling tasks, a systems engineering module (e.g., software module) configured to perform systems engineering tasks, etc. In some embodiments, the STS software modules described herein can reduce the time spent in the spacecraft formulation and design phases of an ESPA-class

spacecraft (i.e., Phases A, B, and the final design sub-phase of Phase C) from 14-30 months to 2 months. In some embodiments, the STS software modules described herein leverage Commercial Off-The-Shelf (COTS) spacecraft subsystem capabilities and enable automated real-time spacecraft systems and computer aided design based on an input of mission requirements.

[0040] In some embodiments, the methods and systems described herein provide an automated design and production process for rapid, modular, technology-refreshable spacecrafts at scale, yielding a spacecraft design process that is fast, relatively inexpensive, modern, and scalable.

[0041] In some embodiments, the methods and systems described herein provide a computer model for mission formulation and a computer model of a spacecraft design. The computer model of a spacecraft design may be referred to herein as a “spacecraft digital twin (or simply ‘digital twin’),” which is a collection of models that represent various aspects (e.g., physical or operation states) of a spacecraft where the digital twin is updated with the current state of a physical twin (also referred to as “hardware twin”), and the physical twin is informed, actuated or influenced by the digital twin. The updates can be performed manually or can be automated.

[0042] In some embodiments, the mission formulation model and the spacecraft digital twin together may enable near real-time tradespace comparison (e.g., analysis of the tradeoffs among resources, costs, and provisioning for different spacecraft designs in the design space). In some embodiments, the mission formulation model provides a digitized, reviewable, and traceable definition of the mission requirements. In some embodiments, the spacecraft digital twin seamlessly interfaces with mission analysis and test models for on-ground operations validation.

[0043] In some embodiments, the methods and systems described herein provide automation of the spacecraft system design process based on digitized mission requirements. System designs may be performance-index optimized, such that the STS software can generate different designs depending on the prioritization or ranking of performance metrics. For example, the STS software can optimize the spacecraft design for a given set of mission requirements to minimize cost if cost is the primary performance metric, or optimize the design for the same set of mission requirements to minimize lead time if lead time is the primary performance metric. Budgets for resources (e.g., mass, power, data, etc.) may be

automatically generated and digitally linked to mission requirements. In some embodiments, automated, traceable, and codified rules-based design evolutions may be provided. For example, the STS software may automatically evolve the design of a given spacecraft to accommodate new, slightly different mission parameters (e.g., a new, slightly different payload) with minimal user input. When the STS software engages in automated design evolution, the software may document the changes in the spacecraft design and the reasons for those changes, so that the changes are visible and directly traceable to the corresponding changes in mission requirements. Such design evolutions may be codified and rule-based in the sense that the changes in spacecraft design are derived from software-based rules (e.g., parametric equations) that relate mission requirements to design specifications.

[0044] In some embodiments, the methods and systems described herein may incorporate Design for Manufacturing and Assembly (DFMA) methodologies (e.g., best practices) and/or Design for Additive Manufacturing (DFAM) methodologies. DFMA refers to the combination of two design methodologies: design for manufacture (e.g., design for ease of manufacture of the parts that will form a product) and design for assembly (e.g., design of the product for ease of assembly). DFMA may be used as the basis for concurrent engineering studies to provide guidance to the design team in simplifying the spacecraft system, reducing manufacturing and assembly costs, and quantifying improvements. DFAM refers to design for manufacturability as applied to Additive Manufacturing (AM), such that functional performance and/or other system life-cycle considerations (e.g., manufacturability, reliability, cost, etc.) are optimized subject to the capabilities of additive manufacturing technologies.

[0045] In some embodiments, DFMA and DFAM methodologies for design and/or delivery of spacecrafts may include automatically determining harness length and path, providing a line of sight to fasteners, and/or providing connector clearances. The combination of modularity, low parts-count, and automated DFMA may enable production and assembly at scale, yielding modular and refreshable spacecraft designs. Also, NASA/JPL have shown that additive manufacturing techniques can enable faster design refresh and added capabilities in the frame/bus. Additive manufacturing techniques may also yield refreshed payload accommodation with minimal changes to the production, assembly, and test processes. In some embodiments, additive manufacturing techniques enable integration of function in the spacecraft frame (e.g., harnessing channels or heat pipes), thereby providing greater spacecraft capability without increasing assembly complexity. By automatically incorporating DFMA and/or DFAM methodologies into the spacecraft design process, the

STS software may enable significant reductions in the duration and expense of spacecraft design and delivery. Furthermore, the STS software's benefits may increase over time, as the software accumulates libraries of previous designs and learns from previous iterations of the design process.

[0046] In some embodiments, the methods and systems described herein incorporate automated process flow and agile-based testing. For example, the methods and systems described herein may include unit-level testing early in the spacecraft manufacturing process, thereby reducing propagation of errors to higher levels of integration which often require more time to troubleshoot. Codified test specifications and sequences derived from mission requirements may be incorporated into the assembly process flow.

[0047] In some embodiments, the methods and systems described herein provide (1) automated estimates of the cost or price of designing and delivering a spacecraft system, (2) automated scheduling of spacecraft assembly, integration, and test activities, and/or (3) identification of longest lead time component.

Some Embodiments of Spacecrafts

[0048] The term "spacecraft" can refer to any human made object configured to orbit a celestial body. Spacecrafts can have a variety of uses, including communication relay, weather forecasting, navigation (e.g., GPS), broadcasting, scientific research, and/or Earth observation. Additional military uses may include reconnaissance, early warning, signals intelligence and weapon delivery. Example spacecrafts include but are not limited to various satellites, space vehicles, etc.

[0049] Most spacecrafts are composed of certain subsystems, for example, communications subsystem, payload subsystem, power subsystem, propulsion subsystem, Attitude Determination and Control System (ADCS) & Guidance, Navigation and Control (GN&C) subsystem, Command and Data Handling (CDH) subsystem. For example, most spacecrafts have an electricity generation system (e.g., solar panels or Radioisotope Thermoelectric Generators (RTGs)) to provide power to on-board equipment. Most spacecrafts have communication equipment (e.g., transponders) suitable for communication with ground stations. Many communication spacecrafts are radio relay stations carrying dozens of transponders, each with a bandwidth of tens of megahertz.

[0050] Some subsystems are made of assemblies, where each assembly includes a certain number of components organized in a specific manner. For example, a power assembly may

include one or more photovoltaic (PV) cells, one or more batteries, and one or more Power Distribution Units (PDUs), and a CDH assembly may include one or more on-board computers (OBCs) and one or more storage components. Under certain circumstances, these assemblies are prebuilt by providers with components provided as a single piece of equipment.

[0051] Many spacecrafts use a standardized bus (e.g., CubeSats). Similar spacecrafts can work together as groups, forming constellations. Because of the high launch cost to space, spacecrafts are generally designed to be as lightweight and robust as possible.

[0052] Orbital launch vehicles carry spacecrafts into space and place them in orbit high enough to avoid orbital decay by the atmosphere. Spacecrafts can then change or maintain the orbit by propulsion, usually by chemical or ion thrusters. Most of the spacecrafts orbiting the Earth are in low Earth orbit or geostationary orbit. Some imaging spacecrafts have a Sun-synchronous orbit so they can scan the entire globe with similar lighting. Most spacecrafts use chemical or ion propulsion to adjust or maintain their orbit, coupled with reaction wheels to control their three axes of rotation or attitude. Without orbit and orientation control, spacecrafts in orbit are generally unable to communicate with ground stations on the Earth.

[0053] ESPA-class spacecrafts are carried by a specific adapter used for launching secondary payloads on orbital launch vehicles. The adapter may be an ESPA adapter, sometimes referred to as an “ESPA ring.” The ESPA utilizes excess launch capacity on an orbital launch vehicle by mounting one or more additional payloads (e.g., ESPA-class spacecrafts) below the primary spacecraft. The use of ESPA-class spacecrafts reduces launch costs for the primary mission and enables secondary and even tertiary spacecrafts with minimal impact on the original mission.

Some Embodiments of Improved Methods for Spacecraft Design

[0054] FIG. 2A shows a conceptual design method 200 of a spacecraft design process, according to some embodiments. In some embodiments, the conceptual design method 200 may be used to perform steps 102-103 of the process 100 for designing and delivering a spacecraft. In some embodiments, an STS system performing the conceptual design method 200 may receive as input data indicating the mission parameters for a spacecraft, and may provide as output conceptual designs for one or more spacecrafts that satisfy the mission parameters. FIG. 3A shows a preliminary design method 300 of a spacecraft design process. In some embodiments, the preliminary design method 300 may be used to perform steps 104-

105 of the process 100 for designing and delivering a spacecraft. In some embodiments, an STS system performing the preliminary design method 300 may receive as input conceptual designs for a spacecraft, and may provide as output preliminary designs for the spacecraft. The conceptual design method 200 and the preliminary design method 300 may be performed by an STS system, which may perform some or all of the tasks generally performed during steps 102-105 of existing implementations of the process 100 (e.g., by systems engineers and/or mechanical engineers).

[0055] In existing implementations of steps 102-103 of the process 100, a systems engineer may spend 2-6 weeks sorting through capabilities (COTS subsystems) and/or writing requirements for the subsystems. The systems engineer generally iterates on this task at least twice for a minimum total of 4-12 weeks of preliminary subsystem selection. The systems engineer may then spend about 2-4 weeks synthesizing mass tables, power tables, block diagrams, and bills of materials.

[0056] Referring to FIG. 2A, a conceptual design method 200 of a spacecraft design process may include steps 202-210. In step 202, the STS system captures one or more mission parameters (or “mission requirements”). In some embodiments, the STS system captures the mission parameters by obtaining (e.g., receiving as input) data indicating a set of spacecraft mission parameters. In some embodiments, the mission parameters are user-specified. In some embodiments, the values of the mission parameters are specified in an input file stemming from a mission design digital twin. In some embodiments, the number of specified mission parameters may be between 3 and 20. Some non-limiting examples of mission parameters may include power, size, mass, communications, pointing, etc.

[0057] In step 204, the STS system selects candidate components for the spacecraft. The candidate components may include candidate subsystems, modules, parts, etc. In some embodiments, the candidate components are selected by a systems engineering module (“SE module”). In some embodiments, the SE module may provide two or more (e.g., 2-4) candidates for each component (including component specifications) at step 204. In some embodiments of step 204, the SE module selects candidate components automatically based on the mission parameters. The specific process of selecting components for different subsystems is further described in detail in FIGS. 4-5D.

[0058] In some embodiments, the SE module includes a data table of spacecraft components and specification parameters (e.g., lead time, price, etc.). In some embodiments,

the SE module includes a data set of specification parameters for each component. In some embodiments, the data set of specification parameters for each component (1) enables mapping of components to mission parameters, (2) enables block diagram generation, (3) enables materials equipment list (MEL) generation, (4) enables power equipment list (PEL) generation, and/or (5) includes volume information, part number information, lead time information, and/or cost information. In some embodiments, the data table of spacecraft components and their specification parameters are updatable on a periodic (e.g., daily, weekly, monthly) basis or as new data becomes available.

[0059] In some embodiments, the SE module includes data codifying relationships between component specification parameters and mission parameters. In some embodiments, the data-codified relationships between component specification parameters and mission parameters enable (1) identification of a set of components that meet mission parameters and (2) identification of a set of components with which mission parameters cannot be achieved. In some embodiments, data codified relationships between specification parameters for different components are provided. In some embodiments, the data codified relationships between specification parameters for different components enable (1) identification of a set of compatible components and (2) identification of specific incompatibilities between components.

[0060] In step 206, the STS system assesses the suitability of one or more sets (e.g., all possible sets) of the candidate components for the mission. As described above, some embodiments of the SE module can ascertain which sets of candidate components do or do not meet the mission parameters based on data codifying relationships between component specifications and mission parameters. In some embodiments of step 206, the SE module selects and keeps track of one or more sets (or combinations) of candidate components that satisfy the specified mission parameters. In some embodiments, the SE module identifies (1) one or more complete sets of components that meet the mission parameters, and/or (2) one or more sets of components that meet the one or more mission parameters and fail to meet one or more other mission parameters. Each set of components that satisfies all the mission parameters may be referred to as a “candidate set of components” or “suitable set of components,” and each set of components that fails to satisfy one or more mission parameters may be referred to as an “unsuitable set of components.” For each unsuitable set of components, the SE module may identify which parameters are satisfied and which parameters are not satisfied by each set of components. In some embodiments, the SE module

may identify a set of candidate components as a suitable set of components only if the components within the set have compatible interfaces or otherwise are compatible with each other.

[0061] In step 208, the STS system identifies one or more candidate spacecraft designs based on the suitable sets of components and on one or more performance metrics of interest. In some embodiments, the SE module analyzes the Pareto efficiency (or “Pareto optimality”) of the suitable sets of components with respect to two or more performance metrics (e.g., schedule, lead time, cost, power, mass, risk, margin of error with respect to mission parameters, etc.). In some embodiments, the SE modules identify any set of components that is on the Pareto frontier for at least two performance metrics of interest as a candidate spacecraft design. Other techniques for identifying one or more of the suitable sets of components as candidate spacecraft designs, or for assessing the optimality of the candidate spacecraft designs, are possible. In some embodiments, the SE module automatically selects sets of spacecraft components that satisfy the mission parameters and minimize (or maximize) performance metrics of interest.

[0062] In step 210, the STS system generates conceptual design plans for each of the candidate spacecraft designs. The conceptual design plan for a candidate spacecraft design may include, for example, a block diagram of the spacecraft design, an orbital model of the spacecraft design, data (e.g., a table) indicating the design’s margins against the mission parameters, a CAD model of the spacecraft design, one or more resource budgets (e.g., mass budget, power budget, communication links budget, etc.) for the design, a Bill of Materials (BoM) for the design, cost or pricing data (e.g., a table) for the design, and/or lead-time data (e.g., a table) for the design. In some embodiments, the CAD model may be generated by a mechanical engineering (ME) module of the STS system. In some embodiments, the orbital model may be generated by an orbital modeling (OM) module of the STS system. In some embodiments, the block diagram, resource budgets, BoM, cost or pricing data, and lead-time data may be generated by the SE module.

[0063] Generation of the CAD model of the spacecraft design by the ME module may include one or more of the following steps. CAD files of the components of the spacecraft design may be imported. In some embodiments, the CAD files for all components are imported into a CAD assembly. In some embodiments, the ME module surveys and catalogs each of the spacecraft’s components to determine whether it is in an enclosure. In some embodiments, the ME module identifies components not contained in enclosures (e.g., solar

arrays, antennas, clamp-band, frame, harnessing, etc.). In some embodiments, the ME module groups components not in enclosures by subsystem. In some embodiments, the ME module automatically designs a rectangular/cubic stack-tray structural architecture for each subsystem group.

[0064] Generation of the CAD model of the spacecraft design by the ME module may further include one or more of the following steps. In some embodiments, the ME module arranges the enclosures. In some embodiments, the enclosures are arranged with minimum clearance between each pair of adjacent enclosures (e.g., 1 inch) into the smallest shape possible that most closely resembles a cube. In some embodiments, two or more alternative versions of the cube are provided. In some embodiments, the ME module adds solar arrays, antennas, and a separation system to the spacecraft design. If the solar arrays are deployable, they may be added to the two opposing faces of the cube. If the solar arrays are body-mounted, they may be added to one or more (e.g., all) otherwise uncovered faces of the cube. In some embodiments, the launch vehicle connecting the clamp-band may be added to the face of the cube not covered by a solar array. In some embodiments, the CAD model produced by the ME module at step 210 visually resembles a spacecraft and is suitable for a rough, order-of-magnitude pricing proposal.

[0065] The conceptual design capabilities of the STS system may scale up over time. For example, the number of candidate components managed by the SE module may increase, which may lead to an increase in the number of sets of candidate components analyzed by the SE module for a mission. As another example, the SE module data codifying relationships between component specification parameters and mission parameters may be refined or enhanced over time, based on user feedback.

[0066] In some embodiments, the resource budgets in a conceptual design may include resource budgets for the mission as a whole and/or for different phases of the mission. In some embodiments, the block diagrams in a conceptual design may be detailed block diagrams (e.g., indicating voltage and data interfaces). In some embodiments, a BoM in a conceptual design may have links to CAD models for the components in the BoM, and the CAD models may be suitable for handoff to a mechanical engineer. In some embodiments, a conceptual design may include metadata (e.g., metadata indicating whether each subsystem/component is provided in a structural enclosure by the supplier). In some embodiments, the conceptual design may be provided in a format suitable for use by a mechanical engineer and/or by government mission analysis and test digital twin software. In

some embodiments, the SE module represents each conceptual design using a data structure suitable for representing spacecraft subsystems. Fields of the data structure may include technical specifications, price, lead time, etc.

[0067] In some embodiments, the STS system may perform the conceptual design method 200 in real time. In some embodiments, prior to or while performing the conceptual design method 200, the STS system may allow the user to specify new components and update the data for components already managed by the SE module (e.g., specify new lead time, cost, or other information for a component).

[0068] In some embodiments, the SE module may accelerate conceptual spacecraft design by providing automated design and cost-estimation capabilities. In some embodiments, as part of step 202 of the conceptual design method 200, the SE module identifies gaps in the specified mission parameters. In some embodiments, the SE module prompts the user to provide inputs that fill the gaps in the specified mission parameters, yielding a complete set of mission parameters.

[0069] In some embodiments, the STS system generates conceptual spacecraft designs in a cloud-based computing environment (e.g., Azure or AWS).

[0070] In some embodiments, the CAD model produced at step 210 visually resembles a spacecraft and is suitable for a Concept of Design Review (CoDR). An example of an excerpt of a CoDR-level design is shown in FIG. 2B.

[0071] Referring to FIG. 3A, a preliminary design method 300 of a spacecraft design process may include steps 302-308. In some embodiments, the preliminary design method 300 may be used to perform steps 104-105 of the process 100 for designing and delivering a spacecraft. One or more steps of the preliminary design method 300 may be performed by a mechanical engineering module (ME module) of the STS system. In some embodiments, the ME module performs some or all of the tasks that a mechanical engineer generally performs during steps 104-105 of existing implementations of the process 100.

[0072] In existing implementations of steps 104-105 of the process 100, a mechanical engineer may spend 4 weeks procuring CAD files for the components of a conceptual spacecraft design and subsequently another 2-4 weeks arranging the parts and designing a frame to support them. Then, after review by a subject matter expert, the mechanical engineer may implement design rules, which generally involve iterating on arranging the components

and designing the frame for another 2-4 weeks. The mechanical engineer may perform these tasks repeatedly depending on review findings.

[0073] In some embodiments, the STS system (e.g., ME module of the STS system) can automatically generate a preliminary spacecraft design based on a conceptual design (e.g., a conceptual design generated using the STS system's conceptual design method 200). In some embodiments, the ME module catalogs all subsystems available for use in the spacecraft design, imports CAD models of spacecraft components into the CAD environment, automatically designs enclosures for components (if required), generates a CAD-based model of the spacecraft frame (and, optionally, optimizes the design of the frame) and combines and arranges the CAD files to generate a preliminary spacecraft design (e.g., a CAD visualization of the spacecraft including spacecraft components, enclosures, and frame).

[0074] In step 302, a conceptual spacecraft design is selected. The spacecraft design may be selected from the set of conceptual spacecraft designs generated using the STS system's conceptual design method 200. In some scenarios, a user selects the spacecraft design and provides input to the STS system indicating the selection. In some scenarios, the STS system selects the spacecraft design based on specified criteria (e.g., a performance metric).

[0075] In step 304, the STS system determines whether any custom interfaces are needed. An interface permits at least one spacecraft component to integrate with another spacecraft component or with a device carried by the spacecraft (e.g., the spacecraft's payload). Some non-limiting examples of interfaces include mechanical interfaces, electrical interfaces, etc. An electrical interface (e.g., breakout board, payload interface board, motherboard, general interface board, etc.) facilitates electrical integration (e.g., coupling or communication) between spacecraft components and/or devices. A mechanical interface (e.g., bracket, hole, joint, etc.) facilitates mechanical integration (e.g., connection or coupling) between spacecraft components and/or devices. The STS system may determine that a custom interface between (or among) spacecraft components/devices is needed if the spacecraft design indicates that an interface between (or among) the components/devices is required and no COTS component suitable for providing such an interface is available. In some embodiments, the STS system uses the ME module to determine whether any custom mechanical interfaces are needed, and uses an electrical engineering (EE) module to determine whether any custom electrical interfaces are needed.

[0076] In step 306, the STS system designs any custom interfaces identified by the system at step 304. In some embodiments, the STS system uses the ME module to design any custom mechanical interfaces and uses the EE module to design any customer electrical interfaces.

[0077] In some embodiments, designing the custom interfaces includes generating the spacecraft frame. In some embodiments, stack-tray structural architecture enclosures for the entire spacecraft (excluding solar, antennas, and clamp-band) are generated. In some embodiments, mass-optimizing features (e.g., thickness variances, x-shaped framing, etc.) are added. For example, in some embodiments, rather than using a stack-tray architecture, generative design is used to produce a mass/load-path optimized frame.

[0078] In step 308, the STS system generates preliminary design plans for the spacecraft. In some embodiments, the STS system updates the conceptual design plans (e.g., CAD model) generated at step 210 of the conceptual design method 200 to incorporate the custom interfaces provided at step 306 of the preliminary design method 300. In some embodiments, the mass of the frame is determined and heuristic rules of the SE module for estimating frame mass are updated accordingly. In some embodiments, the enclosure arrangement design rules may be re-applied (e.g., by the ME module). For example, the enclosure arrangement design rules may be re-applied to update the design to include connector keep-out zones, cable routing constraints, access hole constraints, assembly clearance constraints, etc. In some embodiments, the updated design produced at step 308 visually resembles a PDR-level design. An example of a view of a PDR-level spacecraft design is shown in FIG. 3B.

[0079] In some embodiments, the STS system can automatically perform a final design method to generate a CDR-level spacecraft design based on a preliminary design (e.g., a preliminary design generated using the STS system's preliminary design method 300). This includes generating a digital twin including a collection of models that represent various components of a spacecraft. FIG. 3C illustrates various outputs of generating a digital twin when different components are sequentially added to the spacecraft design based on the conceptual design plan.

[0080] In some embodiments, performing the final design method involves optimizing the frame and applying DFMA rules and/or DFAM rules to the frame's design. In some embodiments, applying DFMA and/or DFAM rules to the frame's design is an iterative process in which DFMA/DFAM rules are sequentially selected; the spacecraft design is checked for compliance with the selected rule; and, if the spacecraft design is not in

compliance with the selected rule, one or more (e.g., all) steps of the conceptual design method 200 and/or the preliminary design method 300 are automatically performed again to update the design to comply with the selected rule. This process of applying DFMA/DFAM rules may continue until the spacecraft design complies with all applicable rules or until the STS system indicates that it is unable to generate a design that complies with one or more of the DFMA/DFAM rules.

Some Specific Implementations in Spacecraft Design

[0081] Referring now to FIG. 4, an exemplary process 400 for selecting candidate components for a spacecraft is provided. The process 400 may be implemented by the disclosed STS system, for example, by the SE module in the system. For example, the SE module may use data codifying relationships between component specification parameters and mission parameters to select specific components included in the subsystems in the spacecraft design, as will be described in detail below. The process 400 will be described in detail with reference to some embodiments illustrated in FIGS. 5A-5D.

[0082] As illustrated in FIG. 4, the process 400 starts with an empty spacecraft 402, which is a default initial state for the spacecraft design. There is no component yet in the empty spacecraft. In the next, a series of subsystem components are sequentially added.

[0083] At step 404, a payload is inserted into the empty spacecraft 402 to get a spacecraft with the payload 406. The processing of inserting the payload includes retrieving a payload profile from a data store storing the payloads and other design-related information or data. Specifically, referring to FIG. 5A, the payload file may include information such as the dimension of the payload (e.g., 20 cm x 20 cm x 20 cm or another different dimension), the mass of the payload (e.g., 10 kg or another different mass), power required by the payload (e.g., 35 W or another different value), transmission requirements (e.g., High or Low). The payload information is stored in a structured database, so that the information for each aspect can be readily identified by the SE module, where the identified information for each aspect is then used in the spacecraft design.

[0084] For example, the information related to the transmission requirement can be used to select the proper communications subsystems in the spacecraft design, as shown in step 408 in FIG. 4. If the transmission requirement is low, UHF and S-Band radios/antenna are then inserted at step 410. On the other hand, if the transmission requirement is medium, UHF,

S, and X-band radios/antennas are inserted at step 412. Similarly, if the transmission requirement is high, UHF, S, and AK-band radios/antennas are inserted at step 414.

[0085] As described earlier, the automated and codified rules-based design is employed by the SE module in the insertion of the communications subsystem in the design. For example, as shown in FIG. 5B, the codified rules implemented by the SE module may include a set of if-then or other forms of rules for selection of proper communications subsystem components. According to one codified rule, the ST module does not insert additional components if the transmission requirement is low, as shown in FIG. 5B. This is because the UHF and S-band radios/antennas are automatically added by default no matter what the transmission requirement is, as also shown in FIG. 5B. On the other hand, when the transmission requirement is not low, additional components are automatically added. For example, according to the codified rules included in the system, when the transmission requirement is high, certain KA-band radios/antennas are then inserted into the spacecraft design, as illustrated in FIG. 5B.

[0086] It should be noted that, when inserting the KA-band radios/antennas, more than one option may be available. Accordingly, more than one spacecraft design can be generated. For example, as illustrated in FIG. 5B, for spacecraft 1, KA-band radio 3 and KA-band antenna 2 are inserted, while for spacecraft 2, KA-band radio 1 and KA-band antenna 1 are inserted into the design. Depending on the availability of KA-band radios/antennas, there may be additional spacecraft designs available, which are not limited in the present disclosure.

[0087] Referring back to FIG. 4, after inserting the communications subsystem at step 410/412/414, a set of spacecraft designs with payload and communications subsystems 416 are obtained. In the next, a propulsion subsystem is added to each spacecraft design at step 418. More specifically, 0-2 thrusters are inserted into each spacecraft design in this step. For example, as shown in FIG. 5C, for the spacecraft design spacecraft 1, a thruster 1 and a thruster 3 are inserted, for another spacecraft design spacecraft 2, 2x thruster 2 are inserted, while for another spacecraft design spacecraft 3, a single thruster 3 is inserted.

[0088] A thruster is a spacecraft propulsion device used for orbital station-keeping, attitude control, or long-duration, low-thrust acceleration, often as part of a reaction control system. Some spacecrafts may not use thrusters but use other propulsion devices such as liquid apogee engine or apogee kick motor. Accordingly, if no thruster is inserted into a

spacecraft design, the SE module may automatically select one of these other propulsion devices for the propulsion subsystem in the spacecraft design. Referring back to FIG. 4, after inserting a propulsion subsystem, a set of spacecraft designs with payload, communications, and propulsion subsystems 420 are obtained.

[0089] In the next, the ADCS and GN&C subsystem is inserted, which includes components used for position determination and attitude determination and for control based on the determinations. Specifically, based on the pointing knowledge requirement determined at step 422, 0-2 star trackers are added at steps 424/426/428. For example, if the pointing knowledge requirement is low, 0 star tracker is added, if the pointing knowledge requirement is medium, 1 star tracker is added, and if the pointing knowledge requirement is high, 2 star trackers are added. Star trackers generally determine the location and attitude of a spacecraft by analyzing the placement of the surrounding stars relative to the payload. If a spacecraft needs to know really well how it is oriented relative to a target, two star trackers may be required. Otherwise, 1 or 0 star tracker is needed for a designed spacecraft.

[0090] At step 430, 1-3 magnetic torquers (or magnetorquers) are added to each spacecraft design. A magnetic torquer is a spacecraft system for attitude control, detumbling, and stabilization built from electromagnetic coils. The number of magnetorquers added depends on how many coils that specific component has. At step 432, 0-4 reaction wheels are added to each spacecraft design. Reaction wheels are essentially flywheels that enable repositioning of controllable space vehicles and spacecrafts while they are in orbit. While reaction wheels are easily exploited because they can generate a torque in any desired direction at any time, magnetorquers rely on a more subtle principle: by generating a magnetic momentum that interacts with the geomagnetic field, a torque is created. In some embodiments, additional ADCS/GN&C components can be added to a spacecraft design. Once these various position and attitude determination and control components are added, a set of spacecraft designs 434 with payload, communications, propulsion, and ADCS/GN&C subsystems are obtained.

[0091] Referring to FIG. 5D, when adding the ADCS/GN&C components 434 to a spacecraft design, for one example spacecraft 1, 1x magnetorquer 2, 3x reaction wheel 1, 2x star tracker 1 are added. For another example spacecraft 2, 3x magnetorquer 1, 4x reaction wheels 3, and 2x star tracker 1 are added. From this and other figures, it can be seen an exponential growth of the number of spacecrafts with every subsystem (or component) added during the evolution of the spacecraft design.

[0092] In the next, the CDH subsystem is added. Specifically, at step 436, a storage device is added to make sure that enough memory is added to satisfy the requirement by the space services department (SSD). In one example, a minimum of 20 GB RAM is required for a spacecraft server to function. In addition, a minimum of 4 GB RAM of swap space is also recommended. Spacecraft running with less RAM than the minimum value might not operate correctly.

[0093] At step 438, an OBC is added. The OBC is the brain of the spacecraft. OBC is responsible for implementation of control law, processing associated with payload, data packeting activities associated with communication, monitoring load health status, handling of data storage, etc. The processor needs to interface with various sensors, actuators present onboard to acquire data to perform its activities and respond accordingly through actuators. Scheduling the activities of the processor is essential due to the number of tasks it has to perform. When adding an OBC, it needs to ensure that the added OBC fulfills the memory requirements and has compatible interfaces with the aforementioned components. Once the CDH subsystem is added, a set of spacecraft designs 440 with payload, communications, propulsion, ADCS/GN&C, and CDH subsystems are obtained.

[0094] In the next, the power subsystem is added. Specifically, at step 442, a set of solar panels are added to provide a mean power consumption for a spacecraft. Typically, 200 to 800 watts of electricity is generated by sunlight and the added solar panels. Depending on the requirement, a higher or lower amount of electricity may be required for a spacecraft, and thus the solar panels added to a design can be adjusted accordingly. At step 444, an appropriate Power Control & Distribution Unit (PCDU) compatible with the selected solar panel generation is added. Once the power system is added, a set of spacecraft designs 446 with payload, communications, propulsion, ADCS/GN&C, CDH, and power subsystems are obtained.

[0095] In the next, the structure subsystem is added to a spacecraft design. The structure subsystem provides the overall mechanical integrity of the spacecraft. It must ensure that all spacecraft components are supported, and that they can withstand handling and launch loads as well as flight in freefall and during operation of propulsive components. To add the structure subsystem, a custom structure is added at step 448, where the added structure may have a mass that is around 80% (or another different value) or the total weight of a designed spacecraft. Once the structure subsystem is added, a set of spacecrafts 450 with payload, communications, ADCS/GN&C, CDH, power, and structure subsystems are obtained.

[0096] At step 452, the obtained set of spacecraft designs may be further filtered to identify the ones that meet the requirements of the mission, which then generates a final list of viable spacecrafts that meet the requirements. In some embodiments, when adding each subsystem or component included in each subsystem, the specific requirements for each subsystem or component included therein may be already taken into account. Accordingly, the step 452 may be not necessarily performed at the end of the design process 400. Although not shown, in some embodiments, the obtained list of viable spacecraft designs may be further ranked according to certain performance metrics (e.g., cost, power, mass, etc.).

[0097] It should be noted that, in the above described process 400, the addition of subsystem components is not limited to the order illustrated in FIG. 4, but can be in other different orders. For example, the propulsion subsystem may be added before the communications subsystem. For another example, reaction wheels may be added before the magnetorquer.

[0098] In some embodiments, based on the components selected in the conceptual design, the STS system further arranges CAD files associated with the selected components to generate a preliminary spacecraft design (e.g., a CAD visualization of the spacecraft including spacecraft components, enclosures, and frame) for each filtered viable spacecraft. In some embodiments, the STS system automatically performs a final design method to generate a CDR-level spacecraft design based on a preliminary design (e.g., a preliminary design generated using the STS system's preliminary design method 300), as described earlier in FIG. 3A, and as further described below in FIG. 6.

System Architecture for Automated Spacecraft Design

[0099] According to some embodiments, the automated spacecraft design may include two different stages, a first stage for automatically selecting a set of subsystems and generating a spacecraft block diagram for a spacecraft design, and a second stage for automatically generating a spacecraft "frame" that will accommodate the set of subsystems. According to some embodiments, the generated "frame" in the second stage of the spacecraft design is a spacecraft CAD model that has a number of specific properties, and the designed frame may be constructed in any of a number (or combination of) traditional or non-traditional manufacturing methods. According to some embodiments, the automated spacecraft structural design disclosed herein may be more related to the second stage of generating a spacecraft CAD model or frame.

[0100] FIG. 6 illustrates a flow diagram of an exemplary automated spacecraft design process 600, according to some embodiments. As illustrated in the figure, the automated spacecraft design process 600 may include a first stage of generating a concept design and a block diagram, including steps as reflected by a dotted box 600a in FIG. 6. Briefly, mission, payload and launch data 605 may be inputs provided by a customer. This information may be used by the automated space vehicle (SV) builder 610, along with the component database 615 to select the subsystems or SV components 620 for the spacecraft. This results in a conceptual design 625, which includes a block diagram, as well as mass and power budgets and a preliminary cost for the spacecraft.

[0101] FIG. 7 illustrates an example block diagram 700 of a concept design of a spacecraft, according to some embodiments. As illustrated in the figure, the block diagram shows a set of subsystems for the designed spacecraft, including sensor subsystem 705 (e.g., star trackers, Global Navigation Spacecraft System (GNSS), sun sensor, etc.), power subsystem 710 (e.g., power conditioning and distribution unit, batteries, solar panels, etc.), control subsystem 715 (e.g., reaction wheels, magnetorquers, thrusters, etc.), and communication subsystem 720 (e.g., S-band, UHF, X-band, etc.). Other components or subsystems included in the concept design may include but are not limited to the payload 725 (which may include scientific or technological instruments carried on board a spacecraft for the specific purpose), interface board 730 (for various subsystems described above), on-board computer 735, and so on. As illustrated, this stage of concept design does not include specific structural details for the designed spacecraft.

[0102] In some embodiments, once customer approval is obtained for the concept mission, the automated spacecraft design process 600 enters into the second stage of the design process, which includes a set of steps and the associated components as indicated by the dotted box 600b in FIG. 6. The second stage 600b of the automated spacecraft design process is more focused on the detailed structural design of the spacecraft, more specifically, the spacecraft frame design, which can be output as a CAD model that is compatible with any of a number of traditional structural design software packages such as Solidworks, AutoCAD, NX, Pro-e, etc. This is labeled as “Parametric/ML Models for Input to High Fidelity Models,” identified by box 660 in FIG. 6. The structural design generated in the software package, is then automatically exported to any of a number of Finite Element Models (FEMs), Finite Difference Models (FDMs) or similar structural analysis packages. This is labeled as ‘High Fidelity Mission Simulations’ indicated by box 655 in FIG. 6. These packages can do either

static load cases or dynamic load cases, such as launch loads or coupled loads analysis as desired. Additional analyses may also be performed, such as, but not limited to, thermal analysis, optical analysis, etc., as further described in detail below.

[0103] It should be noted that, in some embodiments, the frame design disclosed herein refers to the individual frame pieces generated, sized and fitted automatically via automated scripts working within the CAD system, not just putting a tube or a box around the components as many other existing spacecraft design solutions do.

[0104] According to some embodiments, the generation of the CAD model of the spacecraft structural design may be implemented by a mechanical engineering (ME) module of the disclosed automated spacecraft design system. In some embodiments, the CAD model may further include a set of CAD files corresponding to the subsystem components included in the concept design. In some embodiments, the CAD files for all subsystem components may be imported into the generated CAD model or frame or form a CAD assembly (or simply CAD model). In some embodiments, the ME module may survey and catalog each of the spacecraft's subsystem components to determine whether it is in an enclosure. In some embodiments, the ME module may identify subsystem components not contained in enclosures (e.g., solar arrays, antennas, clamp-band, frame, harnessing, etc.). In some embodiments, the ME module may automatically design a rectangular/cubic stack-tray structural architecture for each subsystem component in generating the frame. In some embodiments, the ME module may support a "motherboard" with some or all of the different subsystems plugging into the motherboard.

[0105] According to some embodiments, the generation of the CAD model for the spacecraft design by the ME module may further include one or more of the following steps. In some embodiments, the ME module may arrange the enclosures. In some embodiments, the enclosures may be arranged with minimum clearance between each pair of adjacent enclosures (e.g., 1 inch) into the smallest shape possible that most closely resembles a cube. In some embodiments, two or more alternative versions of the cube may be provided. In some embodiments, the ME module may add solar arrays, antennas, and a separation system to the spacecraft design. If the solar arrays are deployable, they may be added to the two opposing faces of the cube. If the solar arrays are body-mounted, they may be added to one or more (e.g., all) otherwise uncovered faces of the cube. In some embodiments, the launch vehicle connecting the clamp-band may be added to the face of the cube not covered by a

solar array. In some embodiments, the CAD model produced by the ME module visually resembles a spacecraft and is suitable for a rough, order-of-magnitude pricing proposal.

[0106] According to some embodiments, the generated spacecraft frame or CAD model may accommodate all of the subsystems defined by the spacecraft design. According to some embodiments, in addition to accommodating all of the subsystems and their mounting points, the generation of the spacecraft frame or CAD model may also accommodate the DFMA module. For example, a computer aided three-dimensional interactive application (CATIA) may be used in the disclosed automated design. CATIA is a full software suite that incorporates CAD, computer-aided engineering (CAE), and computer-aided manufacturing (CAM).

[0107] According to some embodiments, the accommodation of DFMA module in generating the spacecraft frame may further include automatic enforcement of a set of heuristic rules configured for the automated spacecraft structural design, as will be described in detail later.

[0108] According to some embodiments, in addition to the DFMA rules incorporated for design, the automated spacecraft structural design system may also have the option to export the structural design to one of many load-path optimization designers for further review, analysis, and/or modification. One example of a load-path optimization designer is Fusion 360, which is a 3D-based modeling software, capable of modeling, simulation, and documentation.

[0109] According to some embodiments, the automated spacecraft structural design may be output as a CAD model that is compatible with any of a number of traditional structural design software packages, such as Solidworks, AutoCAD, NX, Pro-e, etc.

[0110] As briefly described earlier, the structural design or the CAD model generated by the ME module of the automated spacecraft design system may also be subject to one or more structural or functional analysis packages for further analysis. For example, the generated CAD model may be automatically exported to any of a number of FEMs, FDMs, or similar structural analysis packages. In some embodiments, these various analysis or simulation packages may conduct analysis on either static load cases or dynamic load cases, such as launch loads or coupled loads analysis as desired. Additional analyses may also be performed, such as, but not limited to, thermal analysis, optical analysis, etc.

[0111] For example, for orbital simulation, a specifically configured System Took Kit (STK) or another different orbital simulation software tool may be used, which models complex systems inside a realistic and time-dynamic three-dimensional simulation that includes high-resolution terrain, imagery, RF environments, and more. STK application allows to select, build, or import precise models of ground, sea, air, and space assets and combine them to represent existing or proposed systems, and allows to simulate the entire system-of-systems in action, at any location and at any time, to gain a clear understanding of its behavior and mission performance. For another example, for structural simulation, Ansys® may be used, which offers structural analysis software solutions that enable engineers of all levels and backgrounds to solve complex structural engineering problems faster and more efficiently. For example, using Ansys, engineers may perform FEA, FDM, customize and automate solutions for structural mechanics challenges, and analyze multiple design scenarios. By using Ansys early in the design cycle, it allows to save costs, and reduce the number of design cycles.

[0112] According to some embodiments, the high fidelity mission simulations also include a thermal simulation. A platform such as SINDA/FLUINT may be used for the desired thermal simulation. SINDA/FLUINT is a comprehensive, generalized tool for simulating complex thermal/fluid systems such as those found in the electronics, automotive, petrochemical, power generation, medical, and aerospace industries. It is a NASA standard software system for thermohydraulic analysis, which provides computational simulation of interacting thermal and fluid effects in designs modeled as heat transfer and fluid flow networks. It is also used to design and analyze aerospace systems. Other thermal simulation tools are also possible, such as Thermal Desktop.

[0113] In some embodiments, during the high fidelity mission simulations, there may be problems, e.g., conflicts found in the arrangements of the selected components in the conceptual design. Accordingly, a constraint optimizer 665 may be used for design optimization. According to some embodiments, the constraint optimizer 665 includes an optimization engine that is configured to use constraint-based planning techniques to generate multiple alternative designs if problems are found in a current design. In one example, the optimization engine included in the constraint optimizer 665 may be configured to prune the design space both in the requirements-to-design direction and possible-design-to-achieve-goals direction, which speeds up the search for candidate alternative components for the subsystems included in a spacecraft design. In some embodiments, the constrain

reasoning/planning-based engine in the constraint optimizer 665 uses procedural constraints, which allows to integrate third-party software into the search process. Such tight integration may allow for earlier identification of possible dead ends, and thus shorter time in the design optimization process.

[0114] According to some embodiments, the outcome of the automated spacecraft design after simulation-based verification and/or constraint-based optimization includes a collection of PDR/CDR-level CAD. In some embodiments, certain assembly instructions accompanying the spacecraft design may also be generated and presented with the PDR/CDR-level CAD.

[0115] According to some embodiments, after the high fidelity mission simulation, a collection of high-fidelity, validated, manufacturable CADs (e.g., ADCS CAD, PV CAD, payload CAD, or all other subsystem CADs) and the system functional model, thermal, structural, and orbital analyses for the specific spacecraft design may form an internal high fidelity digital model 670, also referred to as digital twin or digital flatsat, which is a virtual representation of a designed spacecraft that is as close as possible to a real spacecraft. The digital twin 670 is thus a virtual representation of a physical asset, system, or process that includes its behaviors, performance, and interactions in real time. It primarily relies on data, simulations, and software modeling to reflect the current state and predict future states.

[0116] In some embodiments, this digital twin 670 (i.e., the high-fidelity digital representation of the spacecraft system, also referred to as “digital flatsat”) may co-evolve with a rapidly integrated benchtop hardware twin 675, which is a specific subset of the digital twin, closely tied to the physical hardware system. It often includes direct integration with the physical hardware through embedded sensors and interfaces, enabling monitoring and control. Compared to the digital twin 670 focused on the simulation, optimization, and prediction of the entire spacecraft or operations, the hardware twin 675 is more focused on the physical monitoring and control of hardware systems.

[0117] In some embodiment, the digital twin 670, with twinned hardware-in-the-loop architecture 675, enables early system prototyping and rapid hardware evolution and automatic updating of the digital twin, thereby mitigating integration challenges later in the assembly, integration, and test (AIT) process. In some embodiments, a customer version of the digital twin is also generated, which is tailored for specific customer requirements based on the digital twin 670. In some embodiments, the customer version of the digital twin also co-evolves with the customer-specific hardware twin to align with their unique needs.

[0118] While not shown, in some embodiments, once the above-described automated analyses of a spacecraft design are complete or close to convergence, the DFMA and DFAM may start, as described earlier. The DFMA module may supply a set of additions and constraints (which may be heuristic rules) that are used to evaluate the structural design against manufacturing and assembly constraints. If the design warrants additive manufacturing, the DFAM module may take the revised structural design, determine the load paths, and proceed to a generative optimizer (e.g., topology optimization tools like nTopology, Fusion 360, or Altair Inspire) that will produce an optimized design, to make it manufacturable by additive manufacturing.

[0119] FIG. 8 illustrates an example outcome 800 of a generative spacecraft design for additive manufacturing, in accordance with some embodiments. In the figure, the design is optimized by the generative optimizer by automatically generating and comparing multiple design options to find one or more ideal, best-fit solutions. This may include quick iterations of hundreds or thousands of possible optimized designs through CAD geometry based simulations, resulting in one or more truly optimized solutions that can be then 3D printed, as shown in FIG. 8.

[0120] According to some embodiments, the optimized CAD model or spacecraft frame (or structure) may be constructed in any of a number (or combination of) traditional or non-traditional manufacturing methods. Example methods may include but are not limited to machining, casting, or any of many three-dimensional printing methods. Example materials that may be used for the manufacturing process may include but are not limited to Al alloys, Ti alloys, Inconel alloys, stainless alloys (e.g. 306, 316, C100, etc. ...), Invar alloys, etc.

[0121] According to some embodiments, a large number of automated spacecraft structural designs may be generated using the automated spacecraft design system combined with any of the analyses disclosed herein.

[0122] According to some embodiments, the automated spacecraft design system may further provide a recommendation of a design robustly optimized for particular customer requirements such as performance, cost, lead time, etc. For example, the automated spacecraft design system may rank the automated spacecraft structural designs based on various parameters and choose one of the highly ranked designs according to the customer requirements, as described in detail later in the tradespace exploration.

Some Embodiments of Integration of Heuristic Rules in Automated Spacecraft Design

[0123] In some embodiments, the methods and systems disclosed herein may allow the automated integration of heuristic rules in the DFMA and DFAM for spacecraft design. For example, various heuristic rules may act as guiding principles to streamline the design process, reduce costs, and ensure manufacturability while maintaining the performance and reliability required in spacecraft design. For example, after a structural design is obtained from the above described design process, the DFMA and DFAM rules are integrated into the design process to enable a streamlined process to create manufacturable, efficient, and high-performance structures.

[0124] During the implementation of the DFMA and DFAM modules or during other stages of the spacecraft design, various heuristic rules may be automatically integrated into the modules (e.g., SE, EE, and ME modules) or software applications used in the design, to codify the streamlined process. For example, a heuristic rule may be integrated into a CAD software (e.g., CATIA, Siemens NX) to ensure payload fits with the launch vehicle dimensions. For another example, another heuristic rule may be integrated into the STK software application to simulate thermal conditions to account for the space environment. In this way, a large variety of heuristic rules may be codified and further integrated into the various modules or software applications employed in the spacecraft design.

[0125] According to some embodiments, the possible heuristic rules for the structural design by the DFMA module may include but are not limited to rules for: making sure there is a way (sequence and path) to put the subsystems into the frame; making sure that the attachment points on the frame match the attachment points on the subsystems; making sure that there is enough room for connectors; providing harnessing paths; providing harnessing hooks, hangers or channels; making sure there is enough room for assembly technicians to get their fingers in to attach or detach connectors; making sure there is enough room for screwdrivers to attach connectors; making sure that the connectors are visible; rounding corners; adding gussets; finding common screw sizes; finding other common fasteners; adding stiffening structures; adding cut-outs; etc. It should be noted that the heuristic rules provided here are for illustrative purposes and not for limitations. There are many other heuristic rules that may be applicable to the DFMA module.

[0126] According to some embodiments, some heuristic rules for structural design may be generated based on the experts' recommendations, such as adding stiffening structures,

adding cut-outs, etc. According to some embodiments, some heuristic rules may be generated based on the rules-of-thumb, such as making sure that the attachment points on the frame match the attachment points on the subsystems; making sure that there is enough room for connectors, etc. Certain heuristic rules may be generated for other different purposes, such as reducing cost, complexity, etc. For example, rules for finding common screw sizes and rules for finding other common fasteners may be configured for reduced complexity and/or cost.

[0127] According to some embodiments, these heuristic rules may be automatically integrated into the DFMA implementation. Accordingly, when the DFMA module implements the structural design process, the various heuristic rules may be automatically implemented, to ensure that the designed spacecraft structure meets the constraints and additions specified in these heuristic rules. For example, when the DFMA module automatically implements a heuristic rule for making sure that the attachment points on the frame match the attachment points on the subsystems, the DFMA module (e.g., implemented through the associated software applications) may check the specific parameters for certain attachment points on the frame as well as the parameter for the attachment points on the corresponding subsystems in the current structural design. For example, a door latch in the current structural design may be checked for its size, which is then compared to the size of a strike plate hole in the frame to make sure the two match each other. Similarly, when the DFMA module automatically implements a heuristic rule for making sure there is enough room for assembly technicians to get their fingers in to attach or detach connectors, the DFMA module may first determine the space required for attaching or detaching connectors. The DFMA module may then determine whether the current structural design provides such a space, for example, whether there is enough space for detachment if the spacecraft is assembled based on the current structural design.

[0128] According to some embodiments, if conflicts occur when the DFMA module implements these heuristic rules, the current structural design may be then optimized (e.g., through a constraint optimizer), which may include certain adjustments to the current structural design without necessarily a tremendous change. For example, if it is found that there is not enough room for assembly technicians to get their fingers in to attach or detach connectors in the current structural design, the connectors may be repositioned, e.g., by moving a little farther away from one edge without affecting the desired functions of these connectors.

[0129] Accordingly, by automatically implementing a set of heuristic rules in the DFMA design stage, the DFMA module may simplify manufacturing and assembly processes, reduces costs and lead times, and enhance reliability through fewer parts and standardized designs, thereby taking the structural design to a level that is optimized for manufacturing and assembly checks, e.g., by the DFAM module as described below.

[0130] According to some embodiments, once the design warrants the manufacturing such as additive manufacturing as described above, the DFAM module may take the revised structural design, determine the load paths, and proceed to a generative optimizer that may produce an optimized design, as described above. During the process, the DFAM module may similarly implement certain heuristic rules that are specifically related to additive manufacturing. Example heuristic rules disclosed herein may include but are not limited to rules for: balancing the structure; adding temporary support structures; checking the limits on feature sizes; checking for unsupported overhangs; determining the build direction and orientation on the build plate; checking build volume and partitioning build if needed; avoiding large planes or gridding them, etc.

[0131] According to some embodiments, the DFAM design may cover the optimization of parts for a variety of requirements. One heuristic rule for balancing the structure may require balancing multiple different parts simultaneously, which may contribute to a smooth additive manufacturing process. Another heuristic rule for checking the limits on feature size may require checking certain limits to make sure that the current additive manufacturing can handle such limits. One of the most important elements of DFAM processing is to know the geometric limitations of the processes. Accordingly, various heuristic rules may be created to check ballpark figures of constraints such as the minimum feature size, maximum overhang angle, and minimum wall thicknesses that a machine can produce, which can be then compared to the limits on feature size in the current design to make sure it can be produced by the current machine.

[0132] In real applications, powder-based metal AM may require the use of additional support material for overhanging part geometries, to anchor the part to the build platform and help with heat dissipation. Accordingly, by implementing a heuristic rule for checking for unsupported overhangs, it may determine whether such overhangs are necessary, and if these are, the design of additional support may also be added to the current design.

[0133] When designing for additive manufacturing, one should always design around the specific orientation in which a part will be printed because part orientation will determine the direction of anisotropy, mechanical properties, surface finish, roundness of holes, support material, etc. By implementing a heuristic rule for determining the build direction and orientation on the build plate, it may ensure a desired print orientation may not affect other design decisions thereafter.

[0134] Under certain circumstances, a part's size may exceed the build volume of a machine, and thus it may need to decompose the part into several small sections to be made separately. A heuristic rule for checking build volume and partitioning build if needed may then allow to redesign the part as small sections that the machine will actually build during the additive manufacturing. Similarly, a heuristic rule for avoiding large planes or gridding them may check whether there is a large plane in the current structure design, and may grid such a plane into smaller sections if there is any.

[0135] It should be noted that the above-described heuristic rules for DFAM design are merely for illustrative purposes and not for limitation. Other heuristic rules not described above may be possible and contemplated by the present disclosure. Once these different heuristic rules are implemented by the DFAM module, it may allow the current design to be further optimized to make it manufacturable by additive manufacturing.

[0136] Accordingly, by automatically implementing a set of heuristic rules in the DFAM design stage, the DFAM module may exploit the full potential of additive manufacturing, reduce mass while maintaining performance, and integrate multiple functionalities into single components, saving space and weight, thereby taking the automated spacecraft design to a level that is optimized to be ready for manufacturing and assembly.

[0137] In some embodiments, the heuristic rules are not just integrated into the DFMA and DFAM design stages, but can be applied to any stage of the spacecraft design. For example, various heuristic rules may address aspects related to mission planning, system engineering, thermal management, propulsion, and operational reliability, etc.

[0138] Exemplarily, in the mission and system design, a heuristic rule may be generated to define a clear, achievable objective to minimize complexity and risks, another heuristic rule may be generated to replace solid materials with lightweight composites where feasible to minimize the mass, another heuristic rule may be generated to use standardized satellite buses for different payloads to create modular systems that can be scaled or adapted for different

missions, and another heuristic rule may be generated to design foldable solar panels to fit within the payload fairing to align the design with launch constraints, etc.

[0139] Exemplarily, in the thermal management, a heuristic rule may be generated to use high-conductivity materials for heat pipes to ensure efficient heat transfer between heat sources and sinks, another heuristic rule may be generated to use multi-layer insulation (MLI) and surface coatings for temperature control to minimize reliance on active thermal systems to save energy and reduce complexity, another heuristic rule may be generated to use phase-change materials to manage temperature spikes during eclipse transitions to account for extreme thermal variations in space environments, etc.

[0140] Exemplarily, in the structural design, a heuristic rule may be generated to design structures to withstand vibrations, shock, and g-forces during launch to optimize for launch loads, another heuristic rule may be generated to incorporate fillets and smooth transitions in load-bearing components to minimize the stress concentrations, another heuristic rule may be generated to include redundant support struts for high-risk load paths, etc.

[0141] Exemplarily, in the design of electrical and power system, a heuristic rule may be generated to use high-efficiency solar panels and lightweight batteries to prioritize energy efficiency, another heuristic rule may be generated to implement safeguards such as current limiters and thermal breakers to protect electrical systems, another heuristic rule may be generated to ensure power generation, storage, and consumption are balanced for all mission phases, etc.

[0142] Exemplarily, in the design of propulsion system, a heuristic rule may be generated to design propulsion systems to maximize thrust efficiency and minimize fuel consumption, another heuristic rule may be generated to minimize the number of valves, seals, and fittings in the fuel system to reduce leak risks, another heuristic rule may be generated to include propulsion capacity for collision avoidance and end-of-life disposal to account for space debris avoidance, etc.

[0143] Exemplarily, in the design of communication and control, a heuristic rule may be generated to include multiple communication pathways to avoid single points of failure, another heuristic rule may be generated to avoid shadowing and ensure unobstructed signal paths to ground stations, another heuristic rule may be generated to use fast and reliable control algorithms to ensure quick response to dynamic conditions to minimize latency in control systems, etc.

[0144] In some embodiments, certain heuristic rules may be further generated to ensure reliability and redundancy. Exemplarily, a heuristic rule may be generated to account for the most extreme conditions the spacecraft may encounter. For instance, a heuristic rule may be generated to test components for radiation hardening against cosmic rays. In another example, a heuristic rule may be generated to include systems to detect and mitigate failures autonomously. For instance, a heuristic rule may be generated to use watchdog timers to reset processors in case of software hang-ups. In yet another example, a heuristic rule may be generated to design critical systems with redundancy to ensure mission continuity. For instance, a heuristic rule may be generated to use dual avionics units that can operate independently.

[0145] In some embodiments, certain heuristic rules may be further generated for testing and validation purposes. In one example, a heuristic rule may be generated to simulate mission conditions during ground testing to identify and resolve issues. In another example, a heuristic rule may be generated to test subsystems independently before full system integration. In yet another example, a heuristic rule may be generated to design systems with operational margins to account for uncertainties.

[0146] In some embodiments, certain heuristic rules may be further generated for operational purposes. In one example, a heuristic rule may be generated to ensure the spacecraft can operate with minimal ground intervention. For instance, a heuristic rule may be generated to include onboard AI for real-time anomaly detection and response. In another example, a heuristic rule may be generated to reserve energy and resources for high-demand phases such as orbit insertion. For instance, a heuristic rule may be generated to suspend non-essential activities during propulsion burns. In yet another example, a heuristic rule may be generated to design systems for safe decommissioning to comply with space debris mitigation guidelines. For instance, a heuristic rule may be generated to include deorbit mechanisms or solar sails for controlled reentry.

[0147] In general, these various heuristic rules across these additional stages ensure that the designed spacecraft is not only manufacturable and efficient but also reliable, safe, and capable of meeting mission objectives under extreme conditions. Together, these principles enable spacecraft to perform successfully while adhering to budgetary, temporal, and regulatory constraints.

[0148] In addition, by automatic integration of heuristic rules into a design process, it allows the automatic design process to focus more on the design itself without requiring a designer to first understand a lot about specific details and knowledge behind some specific details in spacecraft design. For example, design engineers with limited experience in the space industry may still be able to implement their concepts in the spacecraft design processes due to the introduction of these heuristic rules, which have incorporated specific knowledge from experts required for the optimal design of spacecraft.

Some Embodiments of Tradespace Exploration in Automated Spacecraft Design

[0149] As described earlier, in some embodiments, the tradespace exploration is included in the spacecraft design disclosed herein. The tradespace in spacecraft design refers to the multidimensional design space where various system parameters, performance metrics, and constraints are analyzed to make informed decisions. It encompasses all potential combinations of design choices, mission objectives, and operational requirements to identify optimal or feasible solutions, or to identify alternative designs without disruption of a design-build process, even if the spacecraft assembly has already been started under certain circumstances. It should be noted that, while the tradespace exploration disclosed herein encompasses all possible combinations, in some embodiments, the tradespace exploration disclosed herein is automatically limited to usable and/or feasible designs, but not to other nonsensical and impractical solutions due to a lack of constraints or validation checks. For example, as will be described later, the design space disclosed herein may be constrained by engineering feasibility rules, ensuring only physically and functionally viable spacecraft configurations are considered.

[0150] In the existing spacecraft design, the tradespace exploration is largely neglected or unexplored, mainly due to the reason that existing spacecraft designs are manually generated. Manually managing a spacecraft design space, or a large number of interactions between the subsystems included in a manual spacecraft design, is quite challenging in the existing spacecraft design. For example, the existing spacecraft design involves a subject matter expert or a team of experts providing a few alternatives that may or may not fulfill mission objectives or close the design. In addition, the existing approaches do not allow for fast re-optimization of the spacecraft design or build if customer objective functions or mission requirements change. That is, the existing approaches are not able to explore the complete design tradespace.

[0151] To address this and other problems in the existing spacecraft design, an automated generation of spacecraft design space (or tradespace) is further provided in the present disclosure. According to embodiments disclosed herein, when the design tradespace is automated, it may capture all possible design choices, which allows many more options (or alternative designs) to be generated, more than humans can reasonably generate or assess. These options or alternative designs may be as good as or better than those generated manually by an expert or teams of experts. Therefore, these design alternatives may become very important alternatives, allowing very convenient re-optimization of a problematic spacecraft design. For example, when a flaw in a particular selected subsystem is identified (e.g., a late Government Industry Data Exchange Program (GIDEP) alert), a set of similar alternatives may be very useful. The automatically generated spacecraft design space may conveniently provide such similar alternatives.

[0152] In other words, through automated tradespace exploration to capture all of the design space, the methods and systems disclosed herein may allow essentially instantaneous selection of alternative designs without disruption of a design-build process, even if the spacecraft assembly has already been started under certain circumstances. The automated generation of spacecraft design tradespace has become feasible due to the introduction of STS system or software as mentioned earlier. For example, by implementing the aforementioned different software modules, the points along the design process where there are design alternatives may be readily identified.

[0153] According to some embodiments, the tradespace for spacecraft design disclosed herein may be considered as a multi-variant mathematical play space used for identifying the optimal boundary spaces (the Pareto frontier) where the multiple variants have strong interdependencies. For example, for a spacecraft tradespace, it may include a large of combinations of subsystems for the generated spacecraft designs. Due to the automation in the design process, the automated spacecraft tradespace may easily capture all reasonable design options, given any automatically generated design space exploration. According to some embodiments, when the design tradespace is automated, many more options can be generated, more than humans can reasonably generate or assess. Often, many alternative designs are as good as or better than those manually generated by an expert or teams of experts.

[0154] According to some embodiments, the automated spacecraft tradespace, when generated, may be specifically tailored to changing customer requirements, to limit the

number of possible combinations or alternatives. For example, some customers favor capacity over schedule, while other customers may favor schedule over capacity. During the automatic design process, these different requirements may be input into the design system as parameters, which then limits certain options taken by the automated design process. Consequentially, the automated design tradespace may also be tailored to changing customer requirements, so as to better serve the customer needs.

[0155] FIG. 9 illustrates an example method 900 for tradespace exploration in spacecraft designs, according to some embodiments. Briefly, the tradespace exploration may include defining the tradespace at step 910, developing models for tradespace evaluation 920, exploring the tradespace at step 930, generating the visualization 940, and further evaluating and refining the spacecraft design at step 950.

[0156] Specifically, to define the tradespace at step 910, the methods and systems disclosed herein may identify design variables, define performance metrics, set constraints, and establish objectives. For example, the design variables may include structural variables, such as material type (e.g., aluminum, titanium), thickness, topology and the like, power variables such as solar panel area, battery capacity, Maximum Power Point Tracking (MPPT) settings and the like, propulsion variables, such as the thrust, ISP (specific impulse), propellant type and the like, and orbit variables such as altitude, inclination, eccentricity and the like. In some embodiments, the tools for setting up the design variables may include but are not limited to MATLAB/Python to set up arrays or parameter sweeps, or parametric CAD tools (e.g., CATIA, Siemens NX) to create adjustable geometries. The performance metrics, on the other hand, may include but are not limited to the total mass, delta-v capability, payload capacity, power availability, cost and other performance metrics, which may be obtained by using tools such as Excel, Python for building metrics calculation spreadsheets, OpenMDAO for performing multidisciplinary performance analysis, etc. The constraints may include but are not limited to budget (e.g., <\$500M), launch vehicle payload capacity (e.g., $\leq 5,000$ kg), environmental limits (e.g., operating temperature -100°C to 100°C), which may be defined in optimization solvers (e.g., `scipy.optimize`, Gurobi) and the like. For objectives, they may include single-objective such as to minimize cost or maximize payload capacity or multi-objective such as balance cost, mass, and delta-v. Possible tools for establishing objectives may include but are not limited to multi-objective optimization frameworks such as NSGA-II, NSGA-III in Python.

[0157] To develop models for tradespace evaluation at step 920, physics-based models, surrogate models, and integrative modeling or other possible modeling may be used. For physics-based models, one or more models may be created to predict performance and interactions among variables. For example, FEA may be explored for load-bearing capacity, thermal desktop may be explored for passive and active thermal control, and STK or General Mission Analysis Tool (GMAT) may be explored for trajectory simulations. For surrogate models, machine learning or simplified models may be used to reduce computational load. For example, polynomial regression may be used to approximate delta-v as a function of propulsion parameters, and Gaussian process regression may be used for thermal response estimation. For integrative modeling, subsystems may be combined for multidisciplinary analysis. For example, OpenMDAO may be used to link propulsion, thermal, structural, and power models, and Simulink may be used for system-level simulation and integration.

[0158] To explore the tradespace at step 930, certain Design of Experiments (DoE), optimization, Monte Carlo simulation, and multi-objective optimization tools may be used. For example, DoE techniques such as full factorial design and Latin hypercube sampling may be used to systematically sample the tradespace. Optimization techniques may be used to identify optimal designs, such as gradient-based optimization for smooth and differentiable tradespace and genetic algorithms for non-linear, discontinuous, or high-dimensional problems. Monte Carlo simulations may be used to perform probabilistic analysis to evaluate the robustness of the spacecraft design under uncertainty, which may include randomly sampling uncertain variables (e.g., thermal loads, material strength), evaluating metrics for each scenario, and analyzing results statistically (e.g., mean, standard deviation). Multi-objective optimization may be used to generate Pareto fronts to identify trade-offs, such as to maximize payload mass while minimizing cost.

[0159] To generate the visualization at step 940, Pareto fronts, heat maps, spider/radar charts, sensitivity analysis and the like may be used. For example, Pareto front may be used to plot trade-offs between two or more objectives (e.g., cost vs. delta-v), heat maps may be used to represent performance metrics across a range of variables such as mass vs. delta-v for different propulsion options, spider/radar charts may be used to compare multiple objectives for a set of spacecraft designs, and sensitivity analysis may be used to visualize the influence of each variable on key metrics.

[0160] To evaluate and refine the spacecraft designs at step 950, certain sensitivity analysis, uncertain analysis, robustness testing and the like may be further conducted. For

example, sensitivity analysis may be used to quantify how changes in variables affect outcomes (e.g., assess how panel area impacts power availability under varying sunlight conditions), uncertainty analysis may be used to account for uncertainties in material properties, launch conditions, etc. (e.g., use probability distributions (e.g., Gaussian, uniform) for uncertain variables), and robustness testing may be used to test candidate designs against worst-case scenarios (e.g., simulate thermal performance under extreme orbital shadowing).

[0161] From the above, it can be seen that the tradespace exploration is a powerful approach to optimizing spacecraft design by systematically analyzing design options, balancing trade-offs, and identifying robust solutions. Leveraging advanced tools, algorithms, and visualization techniques enables informed decision-making across the design lifecycle. In the following, specific details for implementing the tradespace exploration in the spacecraft design is further described.

[0162] As shown in FIG. 10, according to some embodiments, the tradespace exploration 1010 in the spacecraft design disclosed herein may include but are not limited to the following: 1) automated selection of spacecraft subsystem design variables 1020, 2) automated selection of frame design variables 1030, 3) automated selection of integrated functional system design variables 1040, and 4) automated selection of harness/interconnect design variables 1050. In some embodiments, the tradespace exploration 1010 in the spacecraft design may also include the automated selection of DFMA/DFAM rules 1060 related to manufacturing and assembly that essentially affect the spacecraft designs 1070.

[0163] Specifically, in the automated selection of spacecraft subsystems, the methods and systems disclosed herein may consider subsystems required in a spacecraft design and available candidate components for various required subsystems. For an exemplary spacecraft design shown in FIG. 6, the required subsystems may include on-board computing and payload interface, power systems, ADCS, propulsion systems, GNC, communication systems, and separation systems as described earlier. For each subsystem, more than one system component may be included. For each subsystem component, more than one candidate product may be available, as shown in FIGS. 5A-5D.

[0164] For example, for the on-board computing system, exemplary candidates may include but are not limited to Xiphos Technologies Q8JS, Endurosat ES OBC I, and AiTech SPO-S. For the payload interface module, exemplary candidates may include but are not limited to Moog PIB Board and NanoAvionics Payload Controller 2.0. For the power system,

it may include power generation system, power distribution and conditioning system, and power storage and battery system. The power generation system may include but is not limited to solar panels, solar cells, body-mounted solar systems or cells, deployable solar panels, cells or flexible sheets, radioisotope thermal generators. Exemplary candidates for the power system may include but are not limited to MMA Design HaWK17A-42 and ExoTerra 75W Fold-Out Solar Array. For the power distribution and conditioning system, exemplary candidates may include but are not limited to GOM Space NanoPower P60, DHV Technology PCDU, and ACC Clyde Space SmallSat PCDU. For the storage and battery system, it may include but is limited to supercapacitors and hybrid battery supercapacitor systems, and exemplary candidates may include but are not limited to GOM Space Nanopower BP8 and Pumpkin Battery Module 2. For the ADCS, it may include but is not limited to reaction wheels and magnetorquers, and exemplary candidates may include but are not limited to Blue Canyon Technologies XACT-100, GOMSpace GST-600, and Sinclair (Rocketlab) RW4. For the propulsion system, thrusters may include but are not limited to electric, monopropellant, bi-propellant, hall, water vapor, cold gas, green propellant, field effect, and electrospray, and exemplary candidates may include but are not limited to Morpheus Go2 and Dawn Aerospace DA18.01 Satdrive. For the GNC system, it may include but is not limited to star trackers, global positioning spacecraft systems and sun sensors, and exemplary candidates may include but are not limited to Sinclare (Rocketlab) ST-16RT2, Ball Aerospace CT-2020 Star Tracker, Leonardo A-STR, New Space Systems NFSS-411, and Novatel 719 GPS. For the communication system, it may include but is not limited to UHF, VHF, S-Band, C-Band, X-Band, Ka-Band, Ku-Band, W band both in transmit and receive modes, and exemplary candidates may include but are not limited to Endurosat UHF Transceiver II, Rocketlab Frontier-S, GOMSpace NanoCom LinkX, and Space Micro microKaTX-300. For the separation system, it may include but is not limited to separation rings and canisters, and exemplary candidates may include but are not limited to Exolaunch CarbonNIX 8in and Rocketlab PSC Lightband.

[0165] From the above, it can be seen, for a conceptual design, many options may be available for each subsystem component. By exploring the design tradespace, many possible combinations may be automatically generated.

[0166] According to some embodiments, when combining different components in the spacecraft design, the methods and systems disclosed herein may also determine the compatibility between different candidate components. For example, certain power systems

are only compatible with some but not all power distribution systems. In this way, a limited number of combinations may be generated. According to some embodiments, each of these possible combinations may be considered as a design choice (or alternative) in tradespace exploration, which provides an opportunity for a customer to select according to customer needs, for example, selecting one which is more cost-effective, or one which requires less lead time, etc.

[0167] According to some embodiments, besides the subsystems included in the conceptual design, the tradespace exploration in spacecraft design may also take into consideration possible spacecraft frame designs and identify possible options or alternatives. For example, for each possible combination of subsystem components, one or more frame designs may be generated based on the selected subsystem components.

[0168] According to some embodiments, spacecraft frame designs may be automatically generated following the process illustrated in FIG. 2 or following other possible automated design systems, such as frames generated by, but not limited to, automated generative or subtractive designs, load path designs, and so on. Here, generative design is a technology in which 3D models are created and optimized by cloud computing and AI. On the other hand, subtractive design is a process of removing imperfections and extraneous parts in order to strengthen the core elements. Load path design is critical to the structural integrity of a spacecraft. A deliberately designed load path ensures that the load of the structure (especially in a launching process or an orbit process) is transferred from one part to another in a safe and efficient manner. According to some embodiments, the tradespace exploration in spacecraft design may take all these automated spacecraft designs into consideration.

[0169] In some embodiments, the tradespace exploration in spacecraft design may also consider spacecraft frame designs that are manually generated, such as designs generated by an expert or a team of experts.

[0170] In some embodiments, any modifications of those automatically or manually generated designs to make them appropriate for printing, casting, machining or any other fabrication method are also considered in the tradespace exploration. In some embodiments, other designs that may affect the spacecraft frame design may also be considered. For example, as will be described later, certain component integration may affect the frame structure design, which is also considered in the tradespace exploration in spacecraft design.

[0171] According to some embodiments, the tradespace exploration in spacecraft design may also take into consideration integrated functional systems included in a spacecraft. Such integrated functional systems may include but are not limited to thermal systems, fuel systems, filters, mechanisms, etc. According to some embodiments, one or more designs for the integration of specific functional systems may be generated for each combination of subsystems and each associated possible frame design for each combination of subsystem components.

[0172] In one specific example, in spacecraft design, thermal management across spacecraft is a challenge under many circumstances. Internal systems in the spacecraft such as flight computers, payload processors, communications systems, etc., may generate heat. In space, there is no air or atmosphere to move the heat away from these heat generators. The spacecraft may only move heat through conduction or radiation. According to some embodiments, heat may be moved by conduction to other parts of the spacecraft through thermal management systems such as heat exchangers or heat pipes where it may be radiated into space through external fins or radiation panels. These thermal management systems are somewhat costly and are difficult to attach to the spacecraft. These thermal management systems may also add significantly to the volume and mass of the spacecraft. Additionally, at the attachment areas, the heat path is disrupted by changes in materials and by the interfaces between the spacecraft and the thermal management devices leading to significant inefficiencies.

[0173] According to some alternative embodiments, thermal management devices, such as heat exchangers and heat pipes as well as radiating surfaces such as fins or corrugations may be integrated into the spacecraft structure. That is, these devices may be integrated into and become part of the structural elements of the spacecraft. Such design may offer benefits including but not limited to better thermal paths from heat sources to radiating elements with reduced numbers of interfaces and changes in materials, more compact systems because of the integration and removal of the attachment areas required to assemble separate systems together, much more complex, and presumably more effective, systems capable of being integrated at no additional cost, multiple thermal strategies capable of being employed simultaneously with no additional cost (i.e. heat pipes, heat exchangers and radiating fins).

[0174] According to some embodiments, by including the different options for thermal management, the tradespace exploration in spacecraft design may maximize the options when integrating the thermal management components, including the possible cost and thermal

management performance of each possible option. These different options may be further combined with other components in spacecraft design, such as the frame structure design. All of these different options introduce alternatives that may cater to different customer needs or different situations. It should be noted that the above description uses the thermal management system as an example. In real applications, many other functional systems capable of being integrated into the spacecraft design may be similarly processed in the tradespace exploration in spacecraft design.

[0175] According to some embodiments, the tradespace exploration in spacecraft design may also take into consideration of automated integration of DFMA and DFAM rules in automated spacecraft design. For example, when applying DFMA and DFAM rules, alternative options may be found for each possible DFMA and DFAM rule to be applied. The tradespace exploration in spacecraft design disclosed herein may add these various possible options to the design tradespace for consideration and/or further combinations with other system components and/or assembly processes.

[0176] According to some embodiments, the tradespace exploration in spacecraft design may also take into consideration of possible interconnection routes and harnessing combinations. Specifically, in automated spacecraft design, many possible routes may be available for the connector cables for connecting different system components. Similarly, many wire harnessing combinations may also be available in assembling wires to large, multi-faced wiring systems. In spacecraft design, the harness may allow complex arrangement of many wires and cables, which is especially useful in spacecraft design due to the limited space in a spacecraft, where the installation of many separate cables or wires may become troublesome. When arranging wires and cables in harnesses, many possibilities may exist, and the spacecraft designs disclosed herein may take into consideration various harnessing options in the design process.

[0177] According to some embodiments, the tradespace exploration in spacecraft design may take into consideration of additional factors not described above in the design tradespace exploration. In actual applications, the spacecraft design tradespace disclosed herein may consider any part of the design process that introduces trade-off as a design variable. After identifying the options for each design variable, the tradespace exploration may include a generation of all possible combinations of different design variables or different subsystems in spacecraft design. For example, all of the above-described different system components may be combined to generate a larger variety of alternatives in tradespace exploration, where

each alternative may be considered as a design choice in spacecraft design. According to some embodiments, by capturing all of the design options, the methods and systems disclosed herein allow essentially instantaneous selection of alternative designs without disruption of the design-build process even if spacecraft assembly has already been started.

[0178] According to some embodiments, the methods and systems disclosed herein may additionally include an evaluation function or software module (also referred to as an evaluation module) configured to evaluate all possible combinations of systems or design choices. In one example, after obtaining all possible combinations of systems, the evaluation module disclosed herein may implement an evaluation function to generate a score or a set of scores for each specific combination.

[0179] According to some embodiments, the evaluation of the combinations of systems may be performed based on different parameters. For example, in one embodiment, the evaluation of the combinations of systems may be performed based on the cost of a system combination. To achieve such an objective, the total cost for each of all possible combinations of systems is determined, for example, by summing up the cost of each component included in the system combination. In some embodiments, the labor cost for assembling all components together in a system combination may also be estimated. In some embodiments, the cost for a system combination may further include a launch cost. Accordingly, after considering these various factors, an overall cost for a generated system combination may be then determined. In some embodiments, the overall cost may be converted to a score, for example, a higher cost may be converted to a lower score, while a lower cost may be converted to a higher score. According to some embodiments, the generated all possible combinations of systems may be further ranked according to the generated score for each combination. For example, a combination of systems with a higher score in cost may be ranked higher. In this way, all possible combinations of systems can be ranked, which may be then provided to customers who may have cost concerns in spacecraft design.

[0180] According to another embodiment, all possible combinations of systems may also be evaluated based on the lead time of each combination. To achieve such an objective, the evaluation module disclosed herein may implement an evaluation function to determine the lead time for each system combination, for example, how long it takes to complete a process from the design to an end spacecraft product (i.e., spacecraft ready to be loaded for launch). The evaluation function may determine the lead time based on the time required for obtaining

currently available components and/or instant manufacturing of components that are not currently available. The evaluation function may also evaluate the time required for assembling the system components together to form the end product. Based on the evaluated time for these various processes, the lead time for a system combination may be then determined. In some embodiments, the determined lead time may also be converted to a score, for example, a longer lead time may be converted to a lower score and a shorter lead time may be converted to a higher score. According to some embodiments, the generated all possible combinations of systems may be further ranked according to the generated score for each combination. For example, a combination of systems with a higher score in cost may be ranked higher. In this way, all possible combinations of systems can be ranked, which may be then provided to customers who may care about the lead time in spacecraft design.

[0181] According to some embodiments, the evaluation module disclosed herein may utilize other different parameters in implementing an evaluation process. These different factors may include but are not limited to performance, down-link, power consumption, or any other factor that may be specified by customers and that may be relevant to the databases available for the evaluation process. The evaluation for these various parameters may be similarly performed by the evaluation function included in the automated design system disclosed herein.

[0182] According to some embodiments, the possible combinations of systems may also be ranked according to more than one factor or parameter. For example, a customer may be concerned about the cost and lead time, but care less about the performance. Accordingly, the valuation module disclosed herein may generate a combined score for each of all possible combinations of systems, for example, by adding up two scores for each system combination. In another example, a customer may be concerned about the cost, lead time, and performance, and thus a combined score for these three parameters may be generated for each system combination.

[0183] According to some embodiments, an overall score that takes into all possible parameters may be generated for the possible combinations of systems. In generating the overall score, each parameter may have the same weight or may have different weights. For example, if a customer cares more about the cost and lead time, these two parameters may have a larger weight when compared to other factors. In some embodiments, a set of scores including an overall score, a score for each parameter, and one or more scores for certain combinations of factors for each system combination, as well as a set of corresponding ranks

may be provided to the customers, allowing them to have a better understanding of all possible combinations of systems, and to make more informed decisions in determining a final spacecraft design for production. According to some embodiments, if there is a requirement for re-optimization of an existing spacecraft design or build if customer objective functions, or mission requirements change, a larger number of design alternatives and their corresponding scores or ranks may allow a designer to make an easy adjustment in the existing design.

Additional Embodiments

[0184] FIG. 11 is a block diagram of an example computer system 1100 that may be used in implementing the technology described in this document. General-purpose computers, network appliances, mobile devices, or other electronic systems may also include at least portions of the system 1100. The system 1100 includes a processor 1110, a memory 1120, a storage device 1130, and an input/output device 1140. Each of the components 1110, 1120, 1130, and 1140 may be interconnected, for example, using a system bus 1150. The processor 1110 is capable of processing instructions for execution within the system 1100. In some implementations, the processor 1110 is a single-threaded processor. In some implementations, the processor 1110 is a multi-threaded processor. In some implementations, the processor 1110 is a programmable (or reprogrammable) general purpose microprocessor or microcontroller. The processor 1110 is capable of processing instructions stored in the memory 1120 or on the storage device 1130.

[0185] The memory 1120 stores information within the system 1100. In some implementations, the memory 1120 is a non-transitory computer-readable medium. In some implementations, the memory 1120 is a volatile memory unit. In some implementations, the memory 1120 is a non-volatile memory unit.

[0186] The storage device 1130 is capable of providing mass storage for the system 1100. In some implementations, the storage device 1130 is a non-transitory computer-readable medium. In various different implementations, the storage device 1130 may include, for example, a hard disk device, an optical disk device, a solid-state drive, a flash drive, or some other large capacity storage device. For example, the storage device may store long-term data (e.g., database data, file system data, etc.). The input/output device 1140 provides input/output operations for the system 1100. In some implementations, the input/output device 1140 may include one or more network interface devices, e.g., an Ethernet card, a

serial communication device, e.g., an RS-232 port, and/or a wireless interface device, e.g., an 802.11 card, a 3G wireless modem, or a 4G wireless modem. In some implementations, the input/output device may include driver devices configured to receive input data and send output data to other input/output devices, e.g., keyboard, printer and display devices 1160. In some examples, mobile computing devices, mobile communication devices, and other devices may be used.

[0187] In some implementations, at least a portion of the approaches described above may be realized by instructions that upon execution cause one or more processing devices to carry out the processes and functions described above. Such instructions may include, for example, interpreted instructions such as script instructions, or executable code, or other instructions stored in a non-transitory computer readable medium. The storage device 1130 may be implemented in a distributed way over a network, for example as a server farm or a set of widely distributed servers, or may be implemented in a single computing device.

[0188] Although an example processing system has been described in FIG. 11, embodiments of the subject matter, functional operations and processes described in this specification can be implemented in other types of digital electronic circuitry, in tangibly-embodied computer software or firmware, in computer hardware, including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Embodiments of the subject matter described in this specification can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions encoded on a tangible nonvolatile program carrier for execution by, or to control the operation of, a data processing apparatus. Alternatively or in addition, the program instructions can be encoded on an artificially generated propagated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal that is generated to encode information for transmission to suitable receiver apparatus for execution by a data processing apparatus. The computer storage medium can be a machine-readable storage device, a machine-readable storage substrate, a random or serial access memory device, or a combination of one or more of them.

[0189] The term “system” may encompass all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. A processing system may include special purpose logic circuitry, e.g., a Field Programmable Gate Array (FPGA), an Application Specific Integrated Circuit (ASIC), or a programmable general purpose microprocessor or microcontroller. A

processing system may include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them.

[0190] A computer program (which may also be referred to or described as a program, software, a software application, a module, a software module, a script, or code) can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages, and it can be deployed in any form, including as a standalone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A computer program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[0191] The processes and logic flows described in this specification can be performed by one or more programmable computers executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA, an ASIC, or a programmable general purpose microprocessor or microcontroller.

[0192] Computers suitable for the execution of a computer program can include, by way of example, general or special purpose microprocessors or both, or any other kind of central processing unit. Generally, a central processing unit will receive instructions and data from a read-only memory or a random access memory or both. A computer generally includes a central processing unit for performing or executing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic disks, magneto optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, e.g., a mobile telephone, a Personal Digital Assistant (PDA), a mobile audio

or video player, a game console, a Global Positioning System (GPS) receiver, or a portable storage device (e.g., a Universal Serial Bus (USB) flash drive), to name just a few.

[0193] Computer readable media suitable for storing computer program instructions and data include all forms of nonvolatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD-ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[0194] To provide for interaction with a user, embodiments of the subject matter described in this specification can be implemented on a computer having a display device, e.g., a Cathode Ray Tube (CRT) or Liquid Crystal Display (LCD) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's user device in response to requests received from the web browser.

[0195] Embodiments of the subject matter described in this specification can be implemented in a computing system that includes a back end component, e.g., a data server, or that includes a middleware component, e.g., an application server, or that includes a front end component, e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the subject matter described in this specification, or any combination of one or more such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication, e.g., a communication network. Examples of communication networks include a Local Area Network (LAN) and a Wide Area Network (WAN), e.g., the Internet.

[0196] The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network.

The relationship between client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship with each other.

[0197] While this specification contains many specific implementation details, these should not be construed as limitations on the scope of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments. Certain features that are described in this specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable sub-combination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a sub-combination or variation of a sub-combination.

[0198] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

[0199] In addition, programs that implement various aspects of some embodiments may be accessed from a remote location (e.g., a server) over a network. Such data and/or programs may be conveyed through any of a variety of machine-readable medium including, but not limited to: magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as CD-ROMs and holographic devices; magneto-optical media; and hardware devices that are specially configured to store or to store and execute program code, such as ASICs, Programmable Logic Devices (PLDs), flash memory devices, and ROM and RAM devices. Some embodiments may be encoded upon one or more non-transitory, computer-readable media with instructions for one or more processors or processing units to cause steps to be performed. It shall be noted that the one or more non-transitory, computer-readable media shall include volatile and non-volatile memory. It shall also be noted that alternative

implementations are possible, including a hardware implementation or a software/hardware implementation. Hardware-implemented functions may be realized using ASIC(s), programmable arrays, digital signal processing circuitry, or the like. Accordingly, the “means” terms in any claims are intended to cover both software and hardware implementations. Similarly, the term “computer-readable medium or media” as used herein includes software and/or hardware having a program of instructions embodied thereon, or a combination thereof. With these implementation alternatives in mind, it is to be understood that the figures and accompanying description provide the functional information one skilled in the art would require to write program code (i.e., software) and/or to fabricate circuits (i.e., hardware) to perform the processing required.

[0200] It shall be noted that some embodiments may further relate to computer products with a non-transitory, tangible computer-readable medium that has computer code thereon for performing various computer-implemented operations. The medium and computer code may be those specially designed and constructed for the purposes of the techniques described herein, or they may be of the kind known or available to those having skill in the relevant arts. Examples of tangible, computer-readable media include, but are not limited to: magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as CD-ROMs and holographic devices; magneto-optical media; and hardware devices that are specially configured to store or to store and execute program code, such as ASICs, PLDs, flash memory devices, and ROM and RAM devices. Examples of computer code include machine code, such as produced by a compiler, and files containing higher level code that is executed by a computer using an interpreter. Some embodiments may be implemented in whole or in part as machine-executable instructions that may be in program modules that are executed by a processing device. Examples of program modules include libraries, programs, routines, objects, components, and data structures. In distributed computing environments, program modules may be physically located in settings that are local, remote, or both.

[0201] One skilled in the art will recognize no computing system or programming language is critical to the practice of the techniques described herein. One skilled in the art will also recognize that a number of the elements described above may be physically and/or functionally separated into sub-modules or combined together.

[0202] In embodiments, aspects of the techniques described herein (e.g., training a risk assessment model, using a risk assessment model to assess the risk associated with a commit

request, performing one or more (e.g., all) of the steps of the methods described herein, etc.) may be implemented using machine learning and/or artificial intelligence technologies.

[0203] “Machine learning” generally refers to the application of certain techniques (e.g., pattern recognition and/or statistical inference techniques) by computer systems to perform specific tasks. Machine learning techniques may be used to build models based on sample data (e.g., “training data”) and to validate the models using validation data (e.g., “testing data”). The sample and validation data may be organized as sets of records (e.g., “observations” or “data samples”), with each record indicating values of specified data fields (e.g., “independent variables,” “inputs,” “features,” or “predictors”) and corresponding values of other data fields (e.g., “dependent variables,” “outputs,” or “targets”). Machine learning techniques may be used to train models to infer the values of the outputs based on the values of the inputs. When presented with other data (e.g., “inference data”) similar to or related to the sample data, such models may accurately infer the unknown values of the targets of the inference data set.

[0204] A feature of a data sample may be a measurable property of an entity (e.g., person, thing, event, activity, etc.) represented by or associated with the data sample. A value of a feature may be a measurement of the corresponding property of an entity or an instance of information regarding an entity. Features can also have data types. For instance, a feature can have an image data type, a numerical data type, a text data type (e.g., a structured text data type or an unstructured (“free”) text data type), a categorical data type, or any other suitable data type. In general, a feature’s data type is categorical if the set of values that can be assigned to the feature is finite.

[0205] As used herein, “model” may refer to any suitable model artifact generated by the process of using a machine learning algorithm to fit a model to a specific training data set. The terms “model,” “data analytics model,” “machine learning model” and “machine learned model” are used interchangeably herein.

[0206] As used herein, the “development” of a machine learning model may refer to construction of the machine learning model. Machine learning models may be constructed by computers using training data sets. Thus, “development” of a machine learning model may include the training of the machine learning model using a training data set. In some cases (generally referred to as “supervised learning”), a training data set used to train a machine learning model can include known outcomes (e.g., labels or target values) for individual data

samples in the training data set. For example, when training a supervised computer vision model to detect images of cats, a target value for a data sample in the training data set may indicate whether or not the data sample includes an image of a cat. In other cases (generally referred to as “unsupervised learning”), a training data set does not include known outcomes for individual data samples in the training data set.

[0207] Following development, a machine learning model may be used to generate inferences with respect to “inference” data sets. For example, following development, a computer vision model may be configured to distinguish data samples including images of cats from data samples that do not include images of cats. As used herein, the “deployment” of a machine learning model may refer to the use of a developed machine learning model to generate inferences about data other than the training data.

[0208] “Artificial intelligence” (AI) generally encompasses any technology that demonstrates intelligence. Applications (e.g., machine-executed software) that demonstrate intelligence may be referred to herein as “artificial intelligence applications,” “AI applications,” or “intelligent agents.” An intelligent agent may demonstrate intelligence, for example, by perceiving its environment, learning, and/or solving problems (e.g., taking actions or making decisions that increase the likelihood of achieving a defined goal). In many cases, intelligent agents are developed by organizations and deployed on network-connected computer systems so users within the organization can access them. Intelligent agents are used to guide decision-making and/or to control systems in a wide variety of fields and industries, e.g., security; transportation; risk assessment and management; supply chain logistics; and energy management. Intelligent agents may include or use models.

[0209] Some non-limiting examples of AI application types may include inference applications, comparison applications, and optimizer applications. Inference applications may include any intelligent agents that generate inferences (e.g., predictions, forecasts, etc.) about the values of one or more output variables based on the values of one or more input variables. In some examples, an inference application may provide a recommendation based on a generated inference. For example, an inference application for a lending organization may infer the likelihood that a loan applicant will default on repayment of a loan for a requested amount, and may recommend whether to approve a loan for the requested amount based on that inference. Comparison applications may include any intelligent agents that compare two or more possible scenarios. Each scenario may correspond to a set of potential values of one or more input variables over a period of time. For each scenario, an intelligent agent may

generate one or more inferences (e.g., with respect to the values of one or more output variables) and/or recommendations. For example, a comparison application for a lending organization may display the organization's predicted revenue over a period of time if the organization approves loan applications if and only if the predicted risk of default is less than 20% (scenario #1), less than 10% (scenario #2), or less than 5% (scenario #3). Optimizer applications may include any intelligent agents that infer the optimum values of one or more variables of interest based on the values of one or more input variables. For example, an optimizer application for a lending organization may indicate the maximum loan amount that the organization would approve for a particular customer.

[0210] As used herein, "data analytics" may refer to the process of analyzing data (e.g., using machine learning models, artificial intelligence, models, or techniques) to discover information, draw conclusions, and/or support decision-making. Species of data analytics can include descriptive analytics (e.g., processes for describing the information, trends, anomalies, etc. in a data set), diagnostic analytics (e.g., processes for inferring why specific trends, patterns, anomalies, etc. are present in a data set), predictive analytics (e.g., processes for predicting future events or outcomes), and prescriptive analytics (processes for determining or suggesting a course of action).

[0211] Data analytics tools are used to guide decision-making and/or to control systems in a wide variety of fields and industries, e.g., security; transportation; risk assessment and management; supply chain logistics; and energy management. The processes used to develop data analytics tools suitable for carrying out specific data analytics tasks generally include steps of data collection, data preparation, feature engineering, model generation, and/or model deployment.

Terminology

[0212] The phrasing and terminology used herein is for the purpose of description and should not be regarded as limiting.

[0213] Measurements, sizes, amounts, and the like may be presented herein in a range format. The description in range format is provided merely for convenience and brevity and should not be construed as an inflexible limitation on the scope of the invention. Accordingly, the description of a range should be considered to have specifically disclosed all the possible subranges as well as individual numerical values within that range. For example, description of a range such as 1-20 meters should be considered to have specifically disclosed subranges

such as 1 meter, 2 meters, 1-2 meters, less than 2 meters, 10-11 meters, 10-12 meters, 10-13 meters, 10-14 meters, 11-12 meters, 11-13 meters, etc.

[0214] Furthermore, connections between components or systems within the figures are not intended to be limited to direct connections. Rather, data or signals between these components may be modified, re-formatted, or otherwise changed by intermediary components. Also, additional or fewer connections may be used. The terms “coupled,” “connected,” or “communicatively coupled” shall be understood to include direct connections, indirect connections through one or more intermediary devices, wireless connections, and so forth.

[0215] Reference in the specification to “one embodiment,” “preferred embodiment,” “an embodiment,” “some embodiments,” or “embodiments” means that a particular feature, structure, characteristic, or function described in connection with the embodiment is included in at least one embodiment of the invention and may be in more than one embodiment. Also, the appearance of the above-noted phrases in various places in the specification is not necessarily referring to the same embodiment or embodiments.

[0216] The use of certain terms in various places in the specification is for illustration purposes only and should not be construed as limiting. A service, function, or resource is not limited to a single service, function, or resource; usage of these terms may refer to a grouping of related services, functions, or resources, which may be distributed or aggregated.

[0217] Furthermore, one skilled in the art shall recognize that: (1) certain steps may optionally be performed; (2) steps may not be limited to the specific order set forth herein; (3) certain steps may be performed in different orders; and (4) certain steps may be performed simultaneously or concurrently.

[0218] The term “approximately”, the phrase “approximately equal to”, and other similar phrases, as used in the specification and the claims (e.g., “X has a value of approximately Y” or “X is approximately equal to Y”), should be understood to mean that one value (X) is within a predetermined range of another value (Y). The predetermined range may be plus or minus 20%, 10%, 5%, 3%, 1%, 0.1%, or less than 0.1%, unless otherwise indicated.

[0219] The indefinite articles “a” and “an,” as used in the specification and in the claims, unless clearly indicated to the contrary, should be understood to mean “at least one.” The phrase “and/or,” as used in the specification and in the claims, should be understood to mean “either or both” of the elements so conjoined, i.e., elements that are conjunctively present in

some cases and disjunctively present in other cases. Multiple elements listed with “and/or” should be construed in the same fashion, i.e., “one or more” of the elements so conjoined. Other elements may optionally be present other than the elements specifically identified by the “and/or” clause, whether related or unrelated to those elements specifically identified. Thus, as a non-limiting example, a reference to “A and/or B”, when used in conjunction with open-ended language such as “comprising” can refer, in one embodiment, to A only (optionally including elements other than B); in another embodiment, to B only (optionally including elements other than A); in yet another embodiment, to both A and B (optionally including other elements).

[0220] As used in the specification and in the claims, “or” should be understood to have the same meaning as “and/or” as defined above. For example, when separating items in a list, “or” or “and/or” shall be interpreted as being inclusive, i.e., the inclusion of at least one, but also including more than one, of a number or list of elements, and, optionally, additional unlisted items. Only terms clearly indicated to the contrary, such as “only one of” or “exactly one of,” or, when used in the claims, “consisting of,” will refer to the inclusion of exactly one element of a number or list of elements. In general, the term “or” as used shall only be interpreted as indicating exclusive alternatives (i.e. “one or the other but not both”) when preceded by terms of exclusivity, such as “either,” “one of,” “only one of,” or “exactly one of.” “Consisting essentially of,” when used in the claims, shall have its ordinary meaning as used in the field of patent law.

[0221] As used in the specification and in the claims, the phrase “at least one,” in reference to a list of one or more elements, should be understood to mean at least one element selected from any one or more of the elements in the list of elements, but not necessarily including at least one of each and every element specifically listed within the list of elements and not excluding any combinations of elements in the list of elements. This definition also allows that elements may optionally be present other than the elements specifically identified within the list of elements to which the phrase “at least one” refers, whether related or unrelated to those elements specifically identified. Thus, as a non-limiting example, “at least one of A and B” (or, equivalently, “at least one of A or B,” or, equivalently “at least one of A and/or B”) can refer, in one embodiment, to at least one, optionally including more than one, A, with no B present (and optionally including elements other than B); in another embodiment, to at least one, optionally including more than one, B, with no A present (and optionally including elements other than A); in yet another embodiment, to at least one,

optionally including more than one, A, and at least one, optionally including more than one, B (and optionally including other elements).

[0222] The use of “including,” “comprising,” “having,” “containing,” “involving,” and variations thereof, is meant to encompass the items listed thereafter and additional items.

[0223] Use of ordinal terms such as “first,” “second,” “third,” etc., in the claims to modify a claim element does not by itself connote any priority, precedence, or order of one claim element over another or the temporal order in which acts of a method are performed. Ordinal terms are used merely as labels to distinguish one claim element having a certain name from another element having a same name (but for use of the ordinal term), to distinguish the claim elements.

[0224] Particular embodiments of the subject matter have been described. Other embodiments are within the scope of the following claims. For example, the actions recited in the claims can be performed in a different order and still achieve desirable results. As one example, the processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In certain implementations, multitasking and parallel processing may be advantageous. Other steps or stages may be provided, or steps or stages may be eliminated, from the described processes. Accordingly, other implementations are within the scope of the following claims.

[0225] It will be appreciated by those skilled in the art that the preceding examples and embodiments are exemplary and not limiting to the scope of the present disclosure. It is intended that all permutations, enhancements, equivalents, combinations, and improvements thereto that are apparent to those skilled in the art upon a reading of the specification and a study of the drawings are included within the true spirit and scope of the present disclosure. It shall also be noted that elements of any claims may be arranged differently including having multiple dependencies, configurations, and combinations.

[0226] Having thus described several aspects of at least one embodiment of this invention, it is to be appreciated that various alterations, modifications, and improvements will readily occur to those skilled in the art. Such alterations, modifications, and improvements are intended to be part of this disclosure, and are intended to be within the spirit and scope of the invention. Accordingly, the foregoing description and drawings are by way of example only.

WHAT IS CLAIMED IS

1. A method for automated spacecraft design, the method comprising:
 - receiving mission parameters provided by a client for spacecraft design;
 - generating one or more concept designs based on the received mission parameters, a concept design including one or more components selected for each subsystem included in a designed spacecraft;
 - for the concept design, automating a generation of a frame design that accommodates the one or more components selected for each subsystem included in the designed spacecraft, the frame design being a Computer Aided Design (CAD) model that includes individual frame pieces generated, sized and fitted automatically via automated scripts compatible with a CAD system, including one or more structural design software applications;
 - performing a series of analyses or simulations to evaluate a performance of the frame design; and
 - generating a digital twin representing a virtual representation of the designed spacecraft based on the generated frame design and the series of analyses and simulations.
2. The method of claim 1, wherein the digital twin includes the CAD model for the frame design, a set of CAD files representing the one or more components selected for each subsystem included in the designed spacecraft, the series of analyses and simulations and on-orbit performance of the designed spacecraft.
3. The method of claim 1, wherein outcomes of the series of analyses or simulations are provided to a constraint optimizer including an optimization engine that is configured to use constraint-based planning techniques to generate one or more alternative frame designs if one or more problems are found in the frame design.
4. The method of claim 1, wherein automating the generation of the frame design comprises automatically implementing one or more heuristic rules in the frame design.
5. The method of claim 4, wherein the one or more heuristic rules are implemented in a process of applying a Design for Manufacturing and Assembly (DFMA) process in the automated spacecraft design.

6. The method of claim 4, wherein the one or more heuristic rules are implemented in a process of applying a Design for Additive Manufacturing (DFAM) process in the automated spacecraft design.
7. The method of claim 1, wherein generating one or more concept designs based on the received mission parameters comprises generating a plurality of concept designs including a plurality of combinations of subsystems for the plurality of concept designs.
8. The method of claim 7, wherein the plurality of combinations of subsystems for the plurality of concept designs are determined by exploring a tradespace configured for the automated spacecraft design, wherein the tradespace exploration is automatically limited to designs constrained by engineering feasibility rules.
9. The method of claim 1, wherein performing the series of analyses or simulations for the CAD model comprises performing one or more of finite element analyses (FEA) or finite difference method (FDM) for the CAD model.
10. The method of claim 1, wherein performing the series of analyses or simulations for the CAD model comprises performing a thermal simulation for the frame design by using one or more thermal simulation tools.
11. The method of claim 1, further comprising generating a hardware twin that is integrated with the generated digital twin.
12. The method of claim 11, wherein the hardware twin co-evolves with the generated digital twin.
13. A system for automated spacecraft design, the system comprising:
 - a processor; and
 - a memory, coupled to the processor, configured to store executable instructions that, when executed by the processor, cause the processor to:
 - receive mission parameters provided by a client for spacecraft design;

generate one or more concept designs based on the received mission parameters, a concept design including one or more components selected for each subsystem included in a designed spacecraft;

for the concept design, automate a generation of a frame design that accommodates the one or more components selected for each subsystem included in the designed spacecraft, the frame design being a Computer Aided Design (CAD) model that includes individual frame pieces generated, sized and fitted automatically via automated scripts compatible with a CAD system, including one or more structural design software applications;

perform a series of analyses or simulations to evaluate a performance of the frame design; and

generate a digital twin representing a virtual representation of the designed spacecraft based on the generated frame design and the series of analyses and simulations.

14. The system of claim 13, wherein the digital twin includes the CAD model for the frame design, a set of CAD files representing the one or more components selected for each subsystem included in the designed spacecraft, the series of analyses and simulations and on-orbit performance of the designed spacecraft.

15. The system of claim 13, wherein outcomes of the series of analyses or simulations are provided to a constraint optimizer including an optimization engine that is configured to use constraint-based planning techniques to generate one or more alternative frame designs if one or more problems are found in the frame design.

16. The system of claim 13, wherein to automate the generation of the frame design, the executable instructions, when executed by the processor, further cause the processor to automatically implement one or more heuristic rules in the frame design.

17. The system of claim 16, wherein the one or more heuristic rules are implemented in a process of applying a DFMA process in the automated spacecraft design.

18. The system of claim 16, wherein the one or more heuristic rules are implemented in a process of applying a DFAM process in the automated spacecraft design.

19. The system of claim 13, wherein to generate one or more concept designs based on the received mission parameters, the executable instructions, when executed by the processor, further cause the processor to generate a plurality of concept designs including a plurality of combinations of subsystems for the plurality of concept designs.

20. The system of claim 19, wherein the plurality of combinations of subsystems for the plurality of concept designs are determined by exploring a tradespace configured for the automated spacecraft design, wherein the tradespace exploration is automatically limited to designs constrained by engineering feasibility rules.