

VALIDATING A DEVICE UNDER TEST USING TOLERANCE STACKING AND MONTE CARLO SIMULATION**CROSS-REFERENCE TO RELATED APPLICATIONS**

[0001] Priority is claimed to United States Provisional Patent Application No. 63/612,086, filed December 19, 2023, the entire contents of which are hereby incorporated by reference herein.

FIELD OF DISCLOSURE

[0002] The present disclosure generally relates to validating a device under test using a testing system, and more specifically validating a device under test of the testing system having a testing component using tolerance stacking techniques to configure a Monte Carlo simulation.

BACKGROUND

[0003] A testing system designed to test a device under test may include multiple components, any of which may affect the ability of the testing system to reliably carry-out tests which produce test data. Due to imperfect manufacturing processes, a component of a testing system may have various tolerances associated with physical dimensions thereof. These differences in physical dimension may have an impact in the test results provided by the testing system when testing a device under test. In some cases, the worst-case impact of these tolerances may be significant enough such that the test results cannot be relied upon to validate the device under test.

[0004] It is often impractical to measure each and every component of each and every testing system to ensure that the changes in physical dimensions within the tolerance bands doesn't materially affect the test results provided by the system under test. Accordingly, in some scenarios, the tolerance differences for the components of the testing systems may cause some test results provided for devices under test to be invalid. Thus, to validate the reliability of test results provided by a plurality of test systems (e.g., to determine the impact of a change of testing system component or arrangement thereof), the time, effort, cost, and energy associated with inspecting the components of each testing system and/or re-running the tests performed therewith may be substantial and unreasonable. Thus, there is a need to probabilistically evaluate the impact of the component tolerances to validate the reliability of test results provided by the testing system for a device under test.

SUMMARY

[0005] In general, the present disclosure is directed to systems and methods for validating a device under test using a testing system having a testing component.

[0006] In one embodiment, a computer-implemented method of validating a device under test using a testing system having a testing component may include (i) obtaining, by one or more processors, a model of the testing component having a tolerance that impacts test results for a test performed by the testing system, wherein the model indicates an absolute component tolerance metric associated with a dimension of the testing component; (ii) determining, by the one or more processors, the absolute component tolerance metric associated with the dimension of the testing component, wherein the absolute component tolerance metric indicates a worst-case tolerance of the testing component with respect to the test results for the test; (iii) determining, by the one or more processors, a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component, wherein (a) the test result delta is indicative of a worst-case change in a test result of the test performed by the testing system; and (b) the test result delta is determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component; (iv) obtaining, by the one or more processors, a historical test results dataset indicative of historical tests performed using the testing system; (v) generating, by the one or more processors, a plurality of adjusted test results datasets, wherein the adjusted test results datasets are generated by adjusting the

historical test results by a factor generated via a Monte Carlo simulation that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta; (vi) determining, by the one or more processors, a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets; and (vii) determining, by the one or more processors, acceptance or failure of the device under test associated with the historical test results dataset based upon the plurality of sets of statistical characteristics.

[0007] In one variation of the computer-implemented method, the test results indicate one or more of an angle, a length, a width, a displacement, or a force associated with a device under test.

[0008] In another variation of the computer-implemented method, the test is a needle extension test and the test result delta is a change in a length of a needle of a device under test.

[0009] In yet another variation of the computer-implemented method, the test is a force test and the test result delta is a change in active force or hold force of the device under test.

[0010] In still another variation, the computer-implemented method may include determining, by the one or more processors, a shape for the distribution function based upon a statistical analysis of the historical test results dataset.

[0011] In another variation, the computer-implemented method may include determining, by the one or more processors, a tolerance function associated with the dimension of the testing component indicated by the model of the testing component; and determining, by the one or more processors, the shape for the distribution function based upon the tolerance function.

[0012] In yet another variation of the computer-implemented method, determining acceptance of the device under test may include determining, by the one or more processors, that a worst performing set of statistical characteristics satisfies acceptance criteria.

[0013] In another variation of the computer-implemented method, the statistical characteristics include one or more of a mean, a median, a mode, a variance, or a standard deviation.

[0014] In yet another variation of the computer-implemented method, the testing component is a candidate component to replace a current component associated with the testing system.

[0015] In still another variation of the computer-implemented method, in response to determining acceptance of the device under test, the computer-implemented method may include updating, by the one or more processors, a build of materials for the testing system to include the testing component.

[0016] In yet another variation of the computer-implemented method, the test is a first test, the test results are first test results, and the test result delta is a first test result delta; the testing system is configured to provide second test results for a second test; and the computer-implemented method may further include determining, by the one or more processors, a second test result delta indicative of a worst-case change in the second test result for the second test.

[0017] In another variation of the computer-implemented method, the distribution function is a first distribution function; and generating the adjusted historical test results dataset may include adjusting, by the one or more processors, (i) historical test results of the first test using the first distribution function and the first test result delta; and (ii) historical test results of the second test using a second distribution function and the second test result delta.

[0018] In still yet another variation of the computer-implemented method, determining acceptance or failure of the device under test may be based upon a worst performing set of statistical characteristics of the plurality of sets of statistical characteristics.

[0019] In one embodiment, a system for validating a device under test using a testing system having a testing component may include one or more processors; and one or more non-transitory memories storing processor-executable instructions that, when executed by the one or more processors, cause the system to: (i) obtain a model of the testing component having a tolerance that impacts test results for a test performed by the testing system; (ii) determine an absolute component tolerance metric associated with a dimension of a testing component of the testing system, wherein the absolute component tolerance metric indicates a worst-case tolerance of the testing component with respect to the test results for the test; (iii) determine a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component, wherein (a) the test result delta is indicative of a worst-case change in a test result of the test performed by the testing system; and (b) the test result delta is determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component; (iv) obtain a historical test results dataset indicative of historical tests performed using the testing system; (v) generate a plurality of adjusted test results datasets, wherein the adjusted test results datasets are generated by adjusting the historical test results by a factor generated via a Monte Carlo simulation that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta; (vi) determine a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets; and (vii) determine acceptance or failure of the device under test associated with the historical test results dataset based upon the plurality of sets of statistical characteristics.

[0020] In one embodiment, a non-transitory computer-readable medium storing processor-executable instructions that, when executed by one or more processors, cause the one or more processors to: (i) obtain a model of the testing component having a tolerance that impacts test results for a test performed by the testing system; (ii) determine an absolute component tolerance metric associated with a dimension of a testing component of the testing system, wherein the absolute component tolerance metric indicates a worst-case tolerance of the testing component with respect to the test results for the test; (iii) determine a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component, wherein (a) the test result delta is indicative of a worst-case change in a test result of the test performed by the testing system; and (b) the test result delta is determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component; (iv) obtain a historical test results dataset indicative of historical tests performed using the testing system; (v) generate a plurality of adjusted test results datasets, wherein the adjusted test results datasets are generated by adjusting the historical test results by a factor generated via a Monte Carlo distribution that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta; (vi) determine a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets; and (vii) determine acceptance or failure of the device under test associated with the historical test results dataset based the plurality of sets of statistical characteristics.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] The skilled artisan will understand that the figures described herein are included for purposes of illustration and are not limiting on the present disclosure. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the present disclosure. It is to be understood that, in some instances, various aspects of the described implementations may be shown exaggerated or enlarged to facilitate an understanding of the described implementations. In the drawings, like reference characters throughout the various drawings generally refer to functionally similar and/or structurally similar components.

[0022] FIG. 1 is a block diagram of an example testing environment that may be used to verify a device under test of a testing system.

- [0023] FIG. 2A is a side view of an example testing system.
- [0024] FIG. 2B is a top view of a centering washer of the testing system.
- [0025] FIG. 2C is a block diagram of an example autoinjector during a needle extension test.
- [0026] FIG. 2D is a block diagram of an example autoinjector during a force test.
- [0027] FIG. 3 is an example historical needle extension test results dataset and example adjusted needle extension test results datasets.
- [0028] FIG. 4 is a flowchart of an example method for validating a device under test.

DETAILED DESCRIPTION

[0029] The various concepts introduced above and discussed in greater detail below may be implemented in any of numerous ways, and the described concepts are not limited to any particular manner of implementation. Examples of implementations are provided for illustrative purposes.

[0030] FIG. 1 depicts a block diagram of an example testing environment 100 for validating a device under test 110 of a testing system 105 having a testing component 108. The testing environment 100 may include a testing system 105 having one or more testing components 108, a device under test 110, a server 115, and a network 140.

[0031] The testing system 105 may be configured to test the device under test (DUT) 110 by performing one or more tests. For example, the testing system 105 may be configured to perform one or more a quality assurance test, stress test, performance test, verification tests, reliability test, force test, or other suitable test. In a biopharmaceutical product manufacturing context, the testing system 105 may be a needle extension, actuation force, and hold force fixture configured to assess DUTs 110 that include needles (e.g., syringes, autoinjectors, etc.). In other embodiments, the DUT 110 may be any physical device such as mechanical device, electronic device, electro-mechanical device, etc., that is assessed by the testing system 105.

[0032] The testing system 105 may include one or more testing components 108, such as mechanical components (e.g., housings, fixtures, assemblies, screws, washers, struts, supports, holders, etc.), electronic components (e.g., sensor, processor, memory, network interface, power source, etc.), and/or any other suitable component(s). The mechanical components may be associated with manufacturing tolerances indicating a range in possible physical dimensions. For example, a circular component (e.g., a washer, a needle shield, etc.) may include a tolerance associated with an inner radius, an inner diameter, an outer radius, an outer diameter, and so on. In these examples, the component tolerances may be provided by a manufacturer of the component 108 (e.g., in component documentation and/or in a computer model of the component), or in any other suitable manner.

[0033] In at least some aspects, the testing system 105 and/or DUT 110 may be configurable according to the test being conducted, such that the test results may be dependent upon the configuration of the testing system 105. In one example, testing the force applied to a component of the DUT 110 may include the DUT 110 being configured and/or positioned within a range of specific angles (e.g., 45 degrees to 55 degrees) with respect to the testing system 105. Similarly, to conduct a force test, the testing system 105 may be configured with force sensors (e.g., strain gauges) to measure force applied when operating, activating, or otherwise interacting the DUT 110. In another example, testing the life of a battery of a DUT 110 may include electrically coupling the DUT 110 with the testing system 105. In this example, the testing system 105 may be configured with electrical sensors (e.g., voltage, current, etc.) to measure electrical characteristics of the battery of the DUT 110.

[0034] The results of the tests conducted via the testing system 105 for each DUT 110 may be stored in a local and/or remote memory, and/or compiled into test results datasets. The compiled test results datasets may include the data from one or more tests performed with respect to the same DUT 110. For example, the datasets may be a vector of test results generated by the testing system 105 for a plurality of different DUTs 110. Accordingly, when generating the test results, the testing system 105 may append an identifier associated with the DUT 110 (e.g., a batch identifier, a lot identifier, a unit identifier, and/or other types of identifiers used in the industry associated with the DUT 110) to the test result. In at least some embodiments, the testing system 105 may automatically test the DUT 110, e.g., via electromechanical components of the testing system 105 configured to automatically manipulate and/or assess the DUT 110 to automatically execute a test. The testing system 105 may be configured to generate test results data for a test in an automated fashion, e.g., via sensors which generate data associated with the test and/or DUT 110 such as data identifying the testing components 108 of the testing system 105, identifying the DUT 110, measurements, audio, images, video, and/or other suitable test results data. In at least some aspects, the test results data may be transmitted to the server 115 via network 140, e.g., for storage, analysis, etc.

[0035] The testing environment 100 may include a server 115 communicatively coupled to the testing system 105. The server 115 may include a single computing device, or multiple computing devices (e.g., one or more servers and one or more client devices) that are either co-located or remote from each other. In the example embodiment shown in FIG. 1, the server 115 includes one or more processors 120, a network interface 122, a display 124, a user input device 126, and a memory 128. In some embodiments, the server 115 includes a portion of the testing system 105.

[0036] Each of the processor(s) 120 may be a programmable microprocessor that executes software instructions stored in the memory 128 to execute some or all of the functions of the server 115 as described herein. Alternatively, one or more of the processor(s) 120 may be other types of processors (e.g., application-specific integrated circuits (ASICs), field-programmable gate arrays (FPGAs), etc.).

[0037] The network interface 122 may include any suitable hardware (e.g., front-end transmitter and receiver hardware), firmware, and/or software configured to use one or more communication protocols to communicate with external devices and/or systems (e.g., the testing system 105, etc.). For example, the network interface 122 may be or include an Ethernet interface. While not shown in FIG. 1, the server 115 may communicate with the testing system 105, and/or with any device(s) that provide an interface between the server 115 and the testing system 105, via a single communication network 140, or via multiple communication networks of one or more types (e.g., one or more wired and/or wireless local area networks (LANs), and/or one or more wired and/or wireless wide area networks (WANs) such as the Internet or an intranet, etc.).

[0038] The display 124 may use any suitable display technology (e.g., LED, OLED, LCD, etc.) to present information to a user, and the user input device 126 may be a keyboard or other suitable input device. In some embodiments, the display 124 and the user input device 126 are integrated within a single device (e.g., a touchscreen display). Generally, the display 124 and the user input device 126 may combine to enable a user to view and/or interact with visual presentations (e.g., graphical user interfaces or displayed information) output by the server 115, e.g., test results, models, etc.

[0039] The memory 128 may include one or more physical memory devices or units containing volatile and/or non-volatile memory, and may include memories located in different computing devices of the server 115. Any suitable memory type or types may be used, such as read-only memory (ROM), solid-state drives (SSDs), hard disk drives (HDDs), and so on. The memory 128 may store test results data, models, instructions of one or more software applications, and/or any other suitable data. The software applications may include a testing application 130 to control the testing system 105 and/or DUT 110, generate test results data, analyze test results data, model one or more testing components 108 of the testing system 105 and/or DUT 110, and/or any other suitable function.

[0040] The server 115 may include, and/or be communicatively coupled to, a database 135. The database 135 may store models for evaluation of the impact of changes within the component tolerance ranges on the test results generated by the testing system 105. Accordingly, the models may indicate physical dimensions and/or tolerance ranges associated with the testing system 105 and/or one or more components thereof and/or the DUTs 110. It should be appreciated that while in some embodiments the term “model” may be computer-assisted design (CAD) or other three-dimensional model that can be input into tolerance stacking software, in other embodiments, the model may be a set of indications that identify a particular physical dimension and a tolerance range. In some embodiments, the database 135 (or a different database that is not depicted) may also store the test result data received from the testing system 105 and/or a plurality of testing systems 105. The database 135 may be maintained in a persistent memory of the memory 128, or in a different persistent memory of the server 115 or another device or system. In some embodiments, the server 115 accesses the database 135 via the network 140 using the network interface 122. The server 115 may include one device or more devices and/or components, and if multiple devices, may be co-located or remotely distributed (e.g., with Ethernet and/or Internet communication between the different devices).

[0041] One or more components of the testing environment 100 may be communicatively coupled via the network 140, which may be and/or include a proprietary network, a secure public internet, a virtual private network, and/or any other type of suitable network (e.g., dedicated access lines, satellite links, cellular data networks, combinations of these, etc.). The network may be a data bus, a wired network (e.g., Ethernet LAN), a wireless network (e.g., Wi-Fi, Bluetooth, cellular) etc. In embodiments where the network 140 comprises the Internet, data communications may take place over the network 140 via an Internet communication protocol.

[0042] In at least some embodiments, the testing environment 100 may conduct one or more tests of the DUT 110 via the testing system 105 to determine the acceptability of the DUTs 110 and/or test results. The tests may generate test results data which is analyzed by the server 115, e.g., via the testing application 130, to determine whether the DUT 110 passes or fails a test, whether one or more testing components 108 of the testing system 105 operates as expected, whether the test results data is reliable, and the like. For example, a force test may be conducted on the DUT 110 via the testing system 105 to determine an amount of force exerted on a component of the DUT 110 to activate the DUT 110 (e.g., an amount of force applied to a plunger of an autoinjector to activate an automated injection control routine). However, if the test results data produced by the testing system 105 are unreliable and/or otherwise defective due to one or more physical dimensions of a testing component 108 having a tolerance range that significantly alters the test results in a worst-case tolerance, the DUT 110 may appear to pass a test when it should not have if the testing system 105 produced reliable test results data, and vice versa.

[0043] As one example, the DUT 110 may be required to be placed perpendicularly with respect to a testing surface of the testing system 105 to measure a length of a component of the DUT 110 (e.g., a needle). If the DUT 110 is not situated properly during the test, the DUT 110 may not be perpendicular, causing the testing system 105 to generate test results that do not reliably reflect the characteristic of the DUT 110 being tested. For example, if the DUT 110 is situated at an angle between 80 and 100 degrees with respect to the testing surface due to a testing component 108 being closer to a tolerance limit in a physical dimension, it may be unknown whether the testing data is reliable. Accordingly, techniques described herein relate to backtesting historical test results to validate whether the test results are still valid even under a worst-case tolerance in the dimension of the testing component 108.

[0044] In some aspects, more or fewer instances of the various components of the testing environment 100 than are shown in FIG. 1 may be included in the testing environment 100 (e.g., two servers 115, ten testing systems 105, a thousand DUTs 110, etc.).

[0045] FIGS. 2A-2C depict a testing system 205 (such as the testing system 105) configured to perform one or more tests to evaluate characteristics of an autoinjector (such as a DUT 110). In one example, the autoinjector 210 is an insulin injector having a needle which pierces a user's skin when a plunger of the autoinjector 210 is depressed to deliver insulin. In other examples, other types of injectors may be evaluated by the testing system 205.

[0046] Starting with FIG. 2A, illustrated is a side view of the testing system 205. The testing system 205 may include testing components such as a testing surface 215 and a centering washer 220 configured to support the autoinjector 210 when testing characteristics of the autoinjector 210. FIG. 2B illustrates a top view of the centering washer 220 which includes a center aperture 230 to aid in positing the autoinjector 210 in the testing system 205 during testing. In at least some aspects, the autoinjector 210 is preferably positioned perpendicularly with respect to the testing surface 215 when seated properly within the center aperture 230 of the centering washer 220.

[0047] A change in the size and/or position of the center aperture 230 of the centering washer 220 (i.e., the dimension of the testing component 108) may affect the seating of the autoinjector 210 within the testing system 205, potentially impacting the reliability of one or more test results provided by the testing system 205. For example, the center aperture 230 of the centering washer 220 may be sized and/or offset such that a lip of the autoinjector 210 rests upon the washer 220 instead of within the center aperture 230 causing the autoinjector 210 to be seated non-perpendicularly within the testing system 205. If the autoinjector 210 is seated non-perpendicularly, test results which depend on the autoinjector 210 being seated perpendicularly may be unreliable. It should be appreciated that not every dimension of the testing system 205 impacts every test performed by the testing system 205. In the instant example, test results which do not depend on the autoinjector 210 being seated perpendicularly may still be reliable. Accordingly, when evaluating the reliability of a test result provided by the testing system 205, each test result may be independently evaluated.

[0048] In the embodiment according to FIG. 2A, the centering washer 220 centering the autoinjector 210 may be an untested washer, such as a candidate centering washer to replace an existing (tested and verified) centering washer. For example, one may perform the disclosed techniques to ensure that a worst-case tolerance of the candidate untested washer still produce results within acceptance criteria. As another example, the centering washer 220 may be unknowingly placed out of position, replaced by an incorrect washer type, and/or be mis-sized such that the center aperture 230 cannot completely accommodate the autoinjector 210 before conducting several tests. The techniques described herein may be used to determine whether the test results datasets generated by those tests are still reliable to avoid rerunning the tests, which may involve unnecessarily expending additional time, resources (e.g., testing devices), money, etc.

[0049] To determine how a tolerance for a dimension of a testing component, such as the diameter of the center aperture 230 of the centering washer 220, may affect the test results of one or more tests of the testing system 205 and/or acceptable operation of the testing system 205, a computing device such as server 115 may obtain a model of the testing component, e.g., from the database 135, from a manufacturer of the component, by generating a model of the testing component, and/or any other suitable manner of obtaining a model of the testing component. The model may be, include, and/or be based upon documentation, measurements, a bill of materials, a digital model (e.g., computer-aided design, virtual model, etc.), and the like. The testing component model may indicate specifications of the component, such as size, weight, dimensions, tolerances, material, etc.

[0050] In at least some aspects, the testing component model may indicate an absolute tolerance metric associated with a dimension of the testing component, although the absolute component tolerance metric may be obtained in other ways (e.g., a database, documentation, etc.). The absolute component tolerance metric may indicate a worst-case tolerance of the testing component as an absolute value, as, in some cases, the worst-case tolerance may be a maximum and/or positive value/metric

associated with a dimension of the testing component, and in other cases may be a minimum and/or negative value/metric associated with a dimension of the testing component. As such, the absolute value/metric may be used to evaluate the magnitude of the change in dimension, regardless of whether the change in dimension of the testing component is positive or negative.

[0051] For example, the centering washer 220 of the testing system 205 may have a nominal center aperture diameter of 50mm (millimeters) with a diameter tolerance of ± 2 mm (absolute component tolerance of 2mm). While the testing system 205 may perform as expected when the centering washer 220 has a center aperture diameter of 50mm, it may not be clear whether the testing system 205 performs as expected when the centering washer 220 has a center aperture diameter of 48mm or 52mm, as indicated by the worst-case tolerance in the testing component model.

[0052] To determine whether the worst-case tolerance for the centering washer 220 impacts the test results for the testing system 205, the server may determine a test result delta indicative of a worst-case change in a test result of a test performed by the testing system 205 when the testing component exhibits the worst-case tolerance for a given dimension (e.g., center aperture 230 diameter). For example, the testing system 205 may be used to conduct a needle extension test of the autoinjector 210 during which the length of a needle housed in the autoinjector 210 is extended and measured.

[0053] FIG. 2C illustrates a side view of how the testing system 205 measures needle extension. As illustrated in FIG. 2C, the extension length of the needle 240 may be measured from the bottom of testing surface 215 to the distal end of the needle 240 along the perpendicular axis. If the autoinjector 210 is seated perpendicularly within the testing system 205, the needle extension length will be properly measured according to the perpendicular length 245. However, when the autoinjector 210 is tilted at a worst performing angle due to the worst-case tolerance of a dimension of the testing component as indicated by the testing component model, the measured extension length of the needle from the bottom of the testing surface 215 may be shorter than if the autoinjector 210 was seated perpendicularly. That is, the measured test result provided by the testing system 205 may not account for the portion of the needle extension above the testing surface 215.

[0054] Turning to FIG. 2D, illustrated is an example of the testing system 205 performing an activation force test associated with a plunger 260 of the autoinjector 210. During the activation force test, the testing system 205 may be configured to determine the force required to depress the plunger 260 (e.g., to activate the needle 240 of the autoinjector 210). In this example, a change in the size and/or position of the center aperture 230 of the washer 220 (i.e., the dimension of the testing component) may affect the seating of the autoinjector 210 within the testing system 205 such that the pressure applied by the testing system 205 does not align with an activation axis of the autoinjector 210 (e.g., the perpendicular axis with respect to the testing surface 215). As a result, additional force may be required to activate the plunger 260. Accordingly, similar techniques described with respect to the needle extension test of FIGS. 2A-2C may be applied to determine whether the tolerance range of the washer 220 impacted the acceptability of test results provided by the testing system 205.

[0055] To this end, the server (e.g., using testing application 130) may perform various techniques to determine the delta (i.e., test result delta) by which the testing result varies based on the testing component exhibiting a worst-case tolerance for a dimension. In more simple testing systems, such as the testing system 205, the test results delta may be derived using trigonometry. For example, the server may first determine an intermediate value (e.g., the angular offset Θ) caused by the worst-case tolerance of the washer 220. Depending on the washer type, the offset angle may be, for example, 0.5° , 1° , 2° , and so on. Based on the offset angle, the server can apply the geometric relationships using the offset angle to derive the delta in the test result values. It should be appreciated that the intermediate value (e.g., the angular offset) may impact the deltas for the various tests supported by the testing system 205 in different ways. For example, the angular offset may cause the needle extension test

to be inaccurate by, for example 0.1mm, 0.2mm, 0.4mm, and so on, and the activation force test to be inaccurate by, for example, 0.002N, 0.003N, 0.004N, and so on.

[0056] In more complex systems, the server may execute tolerance stacking software that accepts the component models of the various components of the testing system 205 (such as the testing components 108) as inputs and is able to automatically generate an output delta in the test value. In these embodiments, the tolerance stacking software is able to combine the dimensional information indicated by the testing component models into a mathematical framework in which different testing components can be assigned different values within their tolerance ranges to determine the impact on the testing values.

[0057] While the foregoing describes how the testing system 205 may be configured to perform needle extension testing and activation testing, those skilled in the art will appreciate that in other embodiments, other testing systems may be used to generate test results data and deltas associated therewith for other types of tests associated other types of testing systems and/or DUTs.

[0058] As described herein, the delta by which the testing result varies based on the testing component exhibiting the worst-case tolerance for a dimension derived via the above-described tolerance stacking techniques may be applied to backtest datasets to ensure compliance with acceptance criteria. Because the relationship between the delta to the acceptance criteria may be nonlinear, one cannot directly calculate the impact of the delta on the acceptance criteria. One such acceptance criteria are those that rely on a Z-value that involves a ratio of mean, standard deviation, and upper/lower specification limits on acceptability. To solve this problem, techniques disclosed herein apply a Monte Carlo simulation to simulate the impact of the tolerance for the dimension of the testing component. Because Monte Carlo simulations stochastically model the impact of the tolerance as derived from the tolerance stacking analysis, the nonlinear relationships can be derived through a statistical analysis of the results of the Monte Carlo simulation. That is, the combination of tolerance stacking techniques to determine how to configure a Monte Carlo simulation with the performance of the Monte Carlo simulation itself overcome the inability to directly compare the delta to the acceptance criteria. It should be appreciated that while the instant disclosure generally refers to Monte Carlo simulation, other similar types of stochastic sampling and/or simulation techniques are also envisioned.

[0059] To set up the Monte Carlo simulation, the server may obtain and backtest a historical test results dataset indicative of historical tests performed using the testing system, such as already-conducted needle extension tests performed using the testing system 205, to confirm that the test results still meet the acceptance criteria even if the testing component exhibited the worst-case tolerance for the dimension. The historical test results dataset may be in memory such as the memory 128, database 135, and/or otherwise available to the server in any suitable manner. To simulate the effect of the tolerance range of the testing component under test, the server may configure the Monte Carlo simulation to generate and store a plurality of adjusted test results datasets in which the historical test results are adjusted using a random value (or "factor") which does not exceed the test result delta. More particularly, the server may configure the Monte Carlo simulation to use a distribution model that produces the factor when sampled during the Monte Carlo simulation. For example, the server may define the maximum of the distribution model to be the maximum delta derived using the above geometric and/or tolerance stacking techniques. Accordingly, for each result in the historical test results dataset, the server may execute the Monte Carlo simulation to sample the distribution model and add the sampled output to a historical test result to generate a randomly altered test result.

[0060] Depending on the scenario, the distribution model may have different shapes. For example, in some situations, the distribution model may be a uniform distribution model, a Gaussian distribution model, a normal distribution model, a lognormal distribution model, a Weibull distribution model, and so on. It should be appreciated that some testing component models include indications of a distribution of part dimensions within the tolerance range. In these embodiments, the distribution model may be selected to have the same shape as the distribution of part dimensions for the testing component under test. In at least some

examples, the server may determine a tolerance function associated with the dimension of the testing component indicated by the model of the testing component, and set the shape for the distribution function based upon the tolerance function included in the testing component model. It should be appreciated that the server may execute different Monte Carlo simulations that sample different distribution functions having different distribution shapes for different tests supported by the testing system 205.

[0061] With simultaneous reference to FIG. 3, illustrated is an example historical needle extension test results dataset 300 and example adjusted needle extension test results datasets 350a, 350b. The test results datasets 300 indicates historical test results for a lot of 8 DUTs (such as the DUTs 110 and/or autoinjectors 210). It should be appreciated that in other examples, the lot of DUTs may have any number of DUTs. The server then generates the adjusted needle extension test results datasets 350a, 350b by executing a Monte Carlo simulation to sample the distribution model to obtain random values that are added to respective test results in the historical needle extension test results dataset 300. Because the server generates a new sample when generating each value in each adjusted dataset, the adjusted value for a given DUT may vary between the adjusted test result datasets. It should be appreciated that while FIG. 3 depicts the server generating two different adjusted datasets, the server may generate any number of adjusted datasets (e.g., 100 datasets, 500 datasets, 1,000 datasets, 2,000 datasets, and so on) by executing the Monte Carlo simulation any number of times.

[0062] To validate the test results, the server may then compare the adjusted historical test results dataset to the acceptance criteria for the test. Because the adjusted historical test results dataset models the differences in part dimensions within the tolerance range, if the adjusted historical test results dataset still complies with the acceptance criteria, then the worst-case tolerance of the testing component dimension may not have a statistically significant impact on the acceptability of the test results. In some embodiments, the server may repeat the process of adjusting the test results in the historical test results dataset by executing the Monte Carlo simulation for a predetermined number of times to generate a distribution of impacts on the acceptability criteria. In these embodiments, the server may utilize the comparison of the worst performing adjusted historical test results dataset to the acceptance criteria to determine whether the tolerance range for the testing component impacts the acceptability of the historical test results. In the scenario depicted in FIG. 3, this would mean that the server accepts the test results based on whether the adjusted historical dataset 350b still satisfies the acceptance criteria.

[0063] The server may then determine one or more statistical characteristics of the adjusted test results datasets, such as the mean, mode, median, standard deviation, variance, or any other suitable statistical characteristic of the adjusted test results. The server may determine a set of statistical characteristics of the plurality of sets of statistical characteristics, such as a worst performing set of statistical characteristics, and further determine acceptance or failure of the testing system based upon the set of statistical characteristics.

[0064] In at least some aspects, the server may calculate a mean and standard deviation of the adjusted test results dataset to calculate a k-value (also referred to as a critical distance) associated with acceptance criteria for a DUT. More particularly, the server may compare upper and lower Z values (a metric that compares mean, standard deviation, and upper/lower specification limits on acceptability) to the k-value to determine whether the test results associated with the historical test results are acceptable, the DUTs associated with the historical test results are acceptable, or any other suitable determination. If the Z-values are greater than the k-value, then the adjusted test results dataset may be deemed acceptable. Accordingly, the original historical test results were not altered in a statistically significant way based on the tolerance range for the testing component, and can still be relied upon for passing the lot of DUTs. On the other hand, if the Z-values are less than the k-value, the adjusted test results may be deemed unacceptable. In this scenario, it is possible that the original historical test results were impacted by the tolerance range for the testing component and the lot of DUTs may warrant further testing to ensure their acceptability. It should be appreciated that in some embodiments, other acceptance criteria may be used.

[0065] The server may repeat the process of generating adjusted test results datasets and comparing the datasets to the acceptance criteria for each test performed using the testing system 205 to determine the acceptability of the lot of DUTs based on the full suite of testing supported by the testing system 205.

[0066] In embodiments where the aforementioned techniques are used to assess the impact of tolerance ranges with a candidate replacement testing component for the testing system 205, in response to determining acceptability of the adjusted test results and/or DUT, the server may update a build of materials for the testing system 205 to include the candidate replacement testing component (e.g., the centering washer 220), generate a report and/or provide a notification indicating an acceptability of the candidate replacement testing component, place an order for the candidate replacement testing component, and the like.

[0067] As one having skill in the art will understand, the techniques or the systems and methods described may be used to generate test results data of multiple tests, determine the validity of test results data of multiple tests, determine acceptance or failure of DUTs, determine acceptance or failure of an associated testing system, and/or other suitable functions.

[0068] Although the examples and embodiments just described include a server carrying out one or more functions, e.g., obtaining models, determining adjusted test values, determining statistical characteristics, etc., any suitable processor may be used to make such determinations, such as a processor of the testing system 205.

[0069] FIG. 4 depicts a flow diagram of an example computer-implemented method 400 for validating a device under test (such as the DUTs 110 or the autoinjector 210) by a testing system (such as the testing systems 105, 205) having a testing component (such as the testing component 108). In at least some aspects, the testing component may be a candidate component to replace a current component associated with the testing system. The computer-implemented method 400 may operate as a stand-alone method, may operate in conjunction with other systems and/or methods described herein, and/or may operate in conjunction with embodiments of at least a portion of the testing environment 100 of FIG. 1, and/or of any one or more components and/or devices related thereto, and/or with other systems, processors, databases and/or devices. For example, the server 100 may execute at least a portion of the computer-implemented method 400. The computer-implemented method 400 may include additional or alternate steps other than those described with respect to FIG. 4, in embodiments.

[0070] The computer-implemented method 400 may include obtaining, by one or more processors, a model of the testing component having a tolerance that impacts test results for a test performed by the testing system (block 410) (such as the needle extension test described with respect to FIGS. 2A-2C or the activation force test described with respect to FIG. 2D). The model may indicate an absolute component tolerance metric associated with a dimension of the testing component. In at least some aspects, the test results may indicate one or more of an angle, a length, a width, a displacement, or a force associated with a device under test.

[0071] The computer-implemented method 400 may include determining, by the one or more processors, the absolute component tolerance metric associated with the dimension of the testing component (block 420). The absolute component tolerance metric may indicate a worst-case tolerance of the testing component with respect to the test results for the test.

[0072] The computer-implemented method 400 may include determining, by the one or more processors, a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component (block 430). The test result delta may be indicative of a worst-case change in a test result of the test performed by the testing system. In at least some aspects of the computer implemented method 400, the test result delta may be determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component.

[0073] The computer-implemented method 400 may include obtaining, by the one or more processors, a historical test results dataset (such as the dataset 300) indicative of historical tests performed using the testing system (block 440). For example, the historical test results dataset may be obtained from the database 135 or another database.

[0074] The computer-implemented method 400 may include generating, by the one or more processors, a plurality of adjusted test results datasets, such as the datasets 350 (block 450). In at least some aspects of the computer-implemented method 400, the adjusted test results datasets may be generated by adjusting the historical test results by a factor generated via a Monte Carlo simulation that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta.

[0075] The computer-implemented method 400 may include determining, by the one or more processors, a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets (block 460). In at least some aspects, the statistical characteristics may include one or more of a mean, a median, a mode, a variance, or a standard deviation.

[0076] The computer-implemented method 400 may include determining, by the one or more processors, acceptance or failure of the device under test associated with the historical test results based upon the plurality of sets of statistical characteristics (block 470). In at least some aspects, determining acceptance or failure of the device under test (block 470) may be based upon a worst performing sets of statistical characteristics (such as the set of characteristics associated with the dataset 350b) of the plurality of sets of statistical characteristics. In at least some aspects, determining acceptance of the device under test (block 470) may include determining, by the one or more processors, that the worst performing set of statistical characteristics satisfies acceptance criteria. In at least some aspects, in response to determining acceptance of the device under test (block 470), the method 400 may include updating, by the one or more processors, a build of materials for the testing system to include the testing component.

[0077] In at least some aspects, the computer-implemented method 400 may include determining, by the one or more processors, a shape for the distribution function based upon a statistical analysis of the historical test results dataset. In at least some aspects, the computer-implemented method 400 may include determining, by the one or more processors, a tolerance function associated with the dimension of the testing component indicated by the model of the testing component, and determining, by the one or more processors, the shape for the distribution function based upon the tolerance function.

[0078] In at least some aspects of the computer-implemented method 400, the test is a needle extension test and the test result delta is a change in a length of a needle (such as the needle 240) of a device under test. In at least some aspects of the computer-implemented method 400, the test is a force test and the test result delta is a change in activation force or hold force of the device under test. In at least some aspects of the computer-implemented method 400, the testing system may be configured to conduct any other suitable test(s).

[0079] In at least some aspects of the computer-implemented method 400, the test is a first test, the test results are first test results, and the test result delta is a first test result delta. The testing system may be configured to provide second test results for a second test, and the computer-implemented method 400 may further include determining, by the one or more processors, a second test result delta indicative of a worst-case change in the second test result for the second test.

[0080] In at least some aspects of the computer-implemented method 400, the distribution function is a first distribution function, and generating the plurality of adjusted historical test results dataset (block 450) may include adjusting, by the one or more processors, (i) historical test results of the first test using the first distribution function and the first test result delta, and (ii) historical test results of the second test using a second distribution function and the second test result delta.

[0081] This disclosure is intended to explain how to fashion and use various embodiments in accordance with the technology rather than to limit the true, intended, and fair scope and spirit thereof. The foregoing description is not intended to be exhaustive or to be limited to the precise forms disclosed. Modifications or variations are possible in light of the above teachings. The embodiment(s) were chosen and described to provide the best illustration of the principle of the described technology and its practical application, and to enable one of ordinary skill in the art to utilize the technology in various embodiments and with various modifications as are suited to the particular use contemplated. All such modifications and variations are within the scope of the embodiments as determined by the appended claims, as may be amended during the pendency of this application for patent, and all equivalents thereof, when interpreted in accordance with the breadth to which they are fairly, legally and equitably entitled.

[0082] Further, although the foregoing text sets forth a detailed description of numerous different embodiments, it should be understood that the scope of the patent is defined by the words of the claims set forth at the end of this patent. The detailed description is to be construed as exemplary only and does not describe every possible embodiment because describing every possible embodiment would be impractical, if not impossible. Numerous alternative embodiments could be implemented, using either current technology or technology developed after the filing date of this patent, which would still fall within the scope of the claims and all equivalents thereof.

WHAT IS CLAIMED IS:

1. A computer-implemented method of validating a device under test using a testing system having a testing component, the computer-implemented method comprising:
 - obtaining, by one or more processors, a model of the testing component having a tolerance that impacts test results for a test performed by the testing system, wherein the model indicates an absolute component tolerance metric associated with a dimension of the testing component;
 - determining, by the one or more processors, the absolute component tolerance metric associated with the dimension of the testing component, wherein the absolute component tolerance metric indicates a worst-case tolerance of the testing component with respect to the test results for the test;
 - determining, by the one or more processors, a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component, wherein
 - the test result delta is indicative of a worst-case change in a test result of the test performed by the testing system; and
 - the test result delta is determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component;
 - obtaining, by the one or more processors, a historical test results dataset indicative of historical tests performed using the testing system;
 - generating, by the one or more processors, a plurality of adjusted test results datasets, wherein the adjusted test results datasets are generated by adjusting the historical test results by a factor generated via a Monte Carlo simulation that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta;
 - determining, by the one or more processors, a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets; and
 - determining, by the one or more processors, acceptance or failure of the device under test associated with the historical test results dataset based upon the plurality of sets of statistical characteristics.
2. The computer-implemented method of claim 1, wherein determining acceptance or failure of the device under test is based upon a worst performing set of statistical characteristics of the plurality of sets of statistical characteristics.
3. The computer-implemented method of claim 1 or 2, wherein the test results indicate one or more of an angle, a length, a width, a displacement, or a force associated with a device under test.
4. The computer-implemented method of any one of claims 1 to 3, wherein the test is a needle extension test and the test result delta is a change in a length of a needle of a device under test.
5. The computer-implemented method of any one of claims 1 to 4, wherein test is a force test and the test result delta is a change in active force or hold force of the device under test.
6. The computer-implemented method of any one of claims 1 to 5, further comprising:
 - determining, by the one or more processors, a shape for the distribution function based upon a statistical analysis of the historical test results dataset.

7. The computer-implemented method of any one of claims 1 to 6, further comprising:
determining, by the one or more processors, a tolerance function associated with the dimension of the testing component indicated by the model of the testing component; and
determining, by the one or more processors, the shape for the distribution function based upon the tolerance function.

8. The computer-implemented method of any one of claims 1 to 7, wherein determining acceptance of the device under test comprises:
determining, by the one or more processors, that the worst performing set of statistical characteristics satisfies acceptance criteria.

9. The computer-implemented method of any one of claims 1 to 8, wherein the statistical characteristics include one or more of a mean, a median, a mode, a variance, or a standard deviation.

10. The computer-implemented method of any one of claims 1 to 9, wherein the testing component is a candidate component to replace a current component associated with the testing system.

11. The computer-implemented method of any one of claims 1 to 10, further comprising:
in response to determining acceptance of the device under test, updating, by the one or more processors, a build of materials for the testing system to include the testing component.

12. The computer-implemented method of any one of claims 1 to 13, wherein:
the test is a first test, the test results are first test results, and the test result delta is a first test result delta;
the testing system is configured to provide second test results for a second test; and
the method further comprises determining, by the one or more processors, a second test result delta indicative of a worst-case change in the second test result for the second test.

13. The computer-implemented method of claim 12, wherein:
the distribution function is a first distribution function; and
generating the plurality of adjusted historical test results dataset comprises adjusting, by the one or more processors, (i) historical test results of the first test using the first distribution function and the first test result delta; and (ii) historical test results of the second test using a second distribution function and the second test result delta.

14. A system for validating a device under test using a testing system having a testing component, the system comprising:
one or more processors; and
one or more non-transitory memories storing processor-executable instructions that, when executed by the one or more processors, cause the system to:
obtain a model of the testing component having a tolerance that impacts test results for a test performed by the testing system;
determine an absolute component tolerance metric associated with a dimension of a testing component of the testing system, wherein the absolute component tolerance metric indicates a worst-case tolerance of the testing component with respect to the test results for the test;

determine a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component, wherein

the test result delta is indicative of a worst-case change in a test result of the test performed by the testing system; and

the test result delta is determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component;

obtain a historical test results dataset indicative of historical tests performed using the testing system;

generate a plurality of adjusted test results datasets, wherein the adjusted test results datasets are generated by adjusting the historical test results by a factor generated via a Monte Carlo simulation that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta;

determine a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets; and

determine acceptance or failure of the device under test associated with the historical test results dataset based upon the plurality of sets of statistical characteristics.

15. The system of claim 14, wherein the test results indicate one or more of an angle, a length, a width, a displacement, or a force associated with a device under test.

16. The system of claims 14 or 15, wherein determining acceptance or failure of the device under test is based upon a worst performing set of statistical characteristics of the plurality of sets of statistical characteristics.

17. The system of any one of claims 14 to 16, wherein the test is a needle extension test and the test result delta is a change in a length of a needle of a device under test.

18. The system of any one of claims 14 to 17, wherein test is a force test, the testing component is a compression surface of the device under test, and the test result delta is a change in active force or hold force of the device under test.

19. The system of any one of claims 14 to 18, further comprising instructions that, when executed, cause the system to:

determine a shape for the distribution function based upon a statistical analysis of the historical test results dataset.

20. The system of any one of claims 14 to 19, further comprising instructions that, when executed, cause the system to:

determine a tolerance function associated with the dimension of the testing component indicated by the model of the testing component; and

determine the shape for the distribution function based upon the tolerance function.

21. The system of any one of claims 14 to 20, wherein to determine acceptance of the device under test, the system further comprises instructions that, when executed, cause the system to:

determine that the worst performing set of statistical characteristics satisfies an acceptance criteria.

22. The system of any one of claims 14 to 21, wherein the statistical characteristics include one or more of a mean, a median, a mode, a variance, or a standard deviation.

23. The system of any one of claims 14 to 22, wherein the testing component is a candidate component to replace a current component associated with the testing system.

24. The system of any one of claims 14 to 23, further comprising instructions that, when executed, cause the system to, in response to determining acceptance of the testing system, update a build of materials for the testing system to include the candidate component.

25. The system of any one of claims 14 to 24, wherein:
the test is a first test, the test results are first test results, and the test result delta is a first test result delta;
the testing system is configured to provide second test results for a second test; and
the system further comprises instructions that, when executed, cause the system to determine a second test result delta indicative of a worst-case change in a test result for the second test.

26. The system of claim 25, wherein:
the distribution function is a first distribution function; and
to generate the adjusted historical test results dataset, the system further comprises instructions that, when executed, cause the system to adjust (i) historical test results of the first test using the first distribution function and the first test result delta; and (ii) historical test results of the second test using a second distribution function and the second test result delta.

27. A non-transitory computer-readable medium storing processor-executable instructions that, when executed by one or more processors, cause the one or more processors to:
obtain a model of the testing component having a tolerance that impacts test results for a test performed by the testing system;
determine an absolute component tolerance metric associated with a dimension of a testing component of the testing system, wherein the absolute component tolerance metric indicates a worst-case tolerance of the testing component with respect to the test results for the test;
determine a test result delta based upon the absolute component tolerance metric associated with the dimension of the testing component, wherein
the test result delta is indicative of a worst-case change in a test result of the test performed by the testing system; and
the test result delta is determined by a modeling operation of the testing system using the worst-case tolerance of the dimension of the testing component;
obtain a historical test results dataset indicative of historical tests performed using the testing system;
generate a plurality of adjusted test results datasets, wherein the adjusted test results datasets are generated by adjusting the historical test results by a factor generated via a Monte Carlo simulation that uses a distribution function and the test result delta, wherein the factor is a random value not exceeding the test result delta;
determine a plurality of sets of statistical characteristics respectively corresponding to the plurality of adjusted test results datasets; and

determine acceptance or failure of the device under test associated with the historical test results dataset based upon the plurality of sets of statistical characteristics.

28. The non-transitory computer-readable medium of claim 27, wherein the test results indicate one or more of an angle, a length, a width, a displacement, or a force associated with a device under test.

29. The non-transitory computer-readable medium of claim 27 or claim 28, wherein determining acceptance or failure of the device under test is based upon a worst performing set of statistical characteristics of the plurality of sets of statistical characteristics.

30. The non-transitory computer-readable medium of any one of claims 27 to 29, further comprising instructions that, when executed by one or more processors, cause the one or more processors to:
determine a shape for the distribution function based upon a statistical analysis of the historical test results dataset.

31. The non-transitory computer-readable medium of any one of claims 27 to 30, further comprising instructions that, when executed by one or more processors, cause the one or more processors to:
determine a tolerance function associated with the dimension of the testing component indicated by the model of the testing component; and
determine the shape for the distribution function based upon the tolerance function.

32. The non-transitory computer-readable medium of any one of claims 27 to 31, wherein to determine acceptance of the device under test, the non-transitory computer-readable medium comprises instructions that, when executed by one or more processors, cause the one or more processors to:
determine that the worst performing set of statistical characteristics satisfies acceptance criteria.

33. The non-transitory computer-readable medium of any one of claims 27 to 32, wherein:
the test is a first test, the test results are first test results, and the test result delta is a first test result delta;
the testing system is configured to provide second test results for a second test; and
the non-transitory computer-readable medium further comprises instructions that, when executed by one or more processors, cause the one or more processors to determine a second test result delta indicative of a worst-case change in a test result for the second test.