# The Design of an Autonomous Biomimetic Quadruped

Aidan M. Gomez

Academic Magnet High School

May 22, 2017

**Table of Contents**

Design of an Autonomous Quadrupedal Robot

This chapter intends to provide an overview of the study in regards to its general content, being the background which mandates its necessity, the general scientific basis of conclusions made by the researcher, and the procedure that the experiment will follow.

## Topic

The field of robotics fundamentally seeks to extend the capabilities of humankind through automation. Robots are designed with the goal of eliminating the need for a human to be present during the course of a given project, freeing up the resources needed by the human to complete the task at hand. To simplify this definition, a robot can be defined as a mechanical system that accomplishes a task typically carried out exclusively by a human.

According to an online Stanford Computer Science educational document (n.d.), the process of developing 'robots' or mechanical replacements for human efforts date backs to 3000 B.C Egypt, where a water-clock was modified to have figurines ring bells to sound out the hours. After this, primitive 'robots' became a driving force behind human industry. However, wasn't until the mid-20th century that the programmable, computer controlled robots we know today emerged. Today, robots are much different than anything else seen before. They take in, process, and act according to vast amounts of information to accomplish tasks with amazing speed and accuracy, like assembling a car.

The robots of the future need to be smarter than ever before. As robotics marches onward, and the domain of robot expands outwards into our homes, onto our streets, and even onto far away planets, they must be suitably prepared for the chaos of the reality, the unpredictability of every single situation. As Yim et al. (2007) details, robots undertaking long term jobs where human upkeep is not viable, such as space exploration, must be able to ensure their own integrity, i.e. have a high degree of robustness and versatility. In other words, they need to be able to adapt and respond to their environment in order to guarantee the highest chance of goal completion.

## Research Problem

In order for robots to automate increasingly complex processes, as they have throughout history, a gap must be bridged between the robotics that creates solutions for relatively singular tasks, and generalized robotics which produces robots that can exist independently of human maintenance while undertaking different tasks in a dynamic environment. The robots of today are typically poorly suited for efficient functionality in the "real world", where the domain is open – what the robot will encounter is unpredictable, chaotic, and events do not present themselves in a consistent, and reliable manner. In many of these real-world situations human lives depend on the performance of the robot.

One of the "Holy Grails" of this field of adaptable robotics is the self-driving car, which has been extensively researched since as early as 1980's. According to Umson (2008) research in these cars failed to produce a vehicle capable of navigating a real urban environment until the 2007 DARPA Urban Challenge, but even the team which completed the course with their car, "Boss", specifically prioritized adhering to the guidelines of the competition rather than developing an "honest" true-to-life self-driving car. Additionally, pedestrian vehicles were not present in the testing environment. This limitation of the Urban Challenge testing environment in addition to the gamification of the build make Boss a less promising route towards a truly adaptable robot. More recently, large corporations such as Google, Tesla, Uber, and even Apple have begun work on self-driving cars, however, both Tesla and Uber have received criticism for traffic incidents, suggesting that these vehicles have not reached the epitome of adaptive navigation.

But self-driving cars are not the only byproduct of adaptable robotics. In fact according to Henderson (2006), many robots designed for real-world, or open domain, as opposed to a confined domain uses, are turning towards this style of building to maximize the robot's potential. Yamauchi (2004) outlines PackBot, IRobot's multi-use military application ground robot. PackBot is designed to be able to carry out as many tasks as possible for the modern infantryman, including casualty retrieval, reconnaissance, and even short term flight, making it a veritable robotic Swiss army knife. Having a single robot capable of many tasks make it an excellent tool, automating many dangerous tasks simultaneously. This ability to handle many different tasks with relatively few resources seems to be the key to implementing adaptable robots successfully.

## Purpose of the Study

The purpose of this study is to evaluate the benefits of biologically inspired, or biomimetic, locomotion and mechanical design in addition to benefits of machine learning inspired path-finding and mapping algorithms. This means both the electrical and mechanical engineering, in addition to the computer science behind the robot will be tested. In order provide direction, the conductors of the research will ask themselves how effective a robot which has integrated this biomimetic locomotion and unique pathfinding approach is at navigating a maze-like obstacle course autonomously. This study will be similar to that of Hutter et al.'s (2012) review of the STARLETH (Springy Tetrapod with Articulated Robotic Legs) robot in that it features an end to end analysis of the robot's construction and features' contributions towards the robot's success. The research, if successful, could potentially provide an example of an inexpensive yet advanced robot that would be, as speculated by Raibert et al., (2008) capable of reaching the additional up to 50% of land-mass unnavigable by wheeled vehicles, as well as serve as the basis for further research in biomimetic robotics.

## Methods and Evaluation

In order to build this adaptive robot, it first must be constructed to match the requirement of biomimetic hardware and machine-learning inspired software. Additionally, the testing environment must be designed and constructed. The testing environment should reflect the chaos of a real environment, so that the shortcomings of the DARPA Urban Challenge 2007 which were previously discussed in relation to Umson's (2008) work are not present in this study. These shortcomings can be defined exactly in our context as only including three real independent variables, the surrounding cars of a specific size, the rules of the road, and the rules of the competition, so that the real world was not sufficiently modelled. The next step will be a rigorous testing of the robot's capability to navigate the testing environment. The robot's ability to navigate this environment will be measured in catastrophic failures, meaning the robot is unable to complete its goal due to system failure, local failures where a component of the robot fails, but it still accomplishes its goal, and total successes, where the robot essentially flawlessly completes its goal. The goal of the robot in this situation is to navigate to a predetermined location from a starting position with no additional knowledge about its milieu. The first situation, a total failure, can be designated a “-1”, as the robot severely underperformed during operation, the second situation can be denoted as “0”, as the goal was accomplished but with an increased potential for future failure, or a “1”, if the goal was accomplished flawlessly. This scale can provide a quantitative medium for observation of our success if one divides the number of -1’s, 0’s, and 1’s by the number of attempts total to determine the success rate to any degree. If the robot performs with an average of above 0, or a successful average, it succeeds, otherwise, it fails. These methods are creative, in that a robot and testing environment are being produced, deductive, in that the robot’s effectiveness inside of the environment is being tested, and quantitative, in that the deductive results are scaled from -1 to 1 in order to represent degrees of accomplishment.

This study’s success will be determined by the ability of the robot within the testing environment, in whatever form that takes. If the robot is able to navigate the environment without flaws (total success) at least 25% of the time, the researchers will know that the concept shows promise and can be expanded upon in the future, satisfying the goal of a robot that could provide a framework or springboard for future research. Knowing the rate of flawless testing runs also answers the researchers’ governing question of effectiveness of the robot’s autonomy, and thus also its promise as a backboard for future research due to its capability, and thus significance. So long as the robot is not modified during the testing process, and the environment reasonably mimics outside influences, the study will be valid.

## Implications

This study has only one real group of implications that are entirely dependent on its success. If the robot is able to navigate the course with a high rate of success, similar robots could also navigate the real world in such a manner. Principles from the design of this robot could be used in applications like industrial robots and self-driving cars that need dynamic navigation to operate. However, if the robot fails to navigate the course during most of the tests, there are no real implications. As there are too many variables to consider, one cannot be sure that the same approach towards an autonomous vehicle retried under different circumstances would certainly fail, especially if the researchers are more experienced or intelligent than the current researcher.

## Significance

This project attempts to address a sort of Holy Grail for robotics and computer science. One of the most famous long term goals of computer science is to beat the Turing Test, a test that must have a computer program be indistinguishable from a human in conversation in order to be passed. This study hopes to create something that would be indistinguishable from a human in its navigational abilities, or at least on par with a human. This would mean that machines could be substituted for humans in situations requiring this navigational ability.

## Applicability

If a machine which could navigate autonomously in an unknown environment is created as a result of this study, the foundations of autonomous navigation built in this research could be expanded into innumerable applications. The chauffeur and driver industry, a workforce with over two hundred thousand jobs, according to the Bureau of Labor Statistics (n.d.), could be almost entirely eliminated, freeing up those human resources for something more critical. Other researchers could lead the charge in expanding the design principles and navigational framework set forth by this study.

Review of Literature Regarding Autonomous Biomimetic Robotics

This review of literature will center on the comparison of studies which focus on components of the design and development of an autonomous quadrupedal robot. These studies can be clearly subdivided into two main categories; those based on the physical structure of the robot, and those which make up the control system. Specifically, research in which falls in the first category, hardware, will involve mimicking the kinematics and gait of a quadrupedal animal, most commonly the dog, through the construction and design of the robot. The second category, software, will cover works which outline the theory of different algorithms which would play a part in the control of the robot, such as pathfinding, localization and mapping, and detection and tracking of obstacles.

## Software

Although the demarcation of software and hardware has increasingly blurred over time with the introduction of complex integrated circuits and logic chips, software within this context will be defined as a program running on an onboard Linux system that controls hardware in some facsimile, and the algorithms that compose it.

### Movement Planning

In order for any mobile robot to navigate to an objective there must be some form of a movement planning algorithm. Defined by Guo and Parker (2002), a motion planning algorithm generates a course of action that achieves a goal while avoiding collisions. Ultimately, the algorithm must derive a map of its environment, localize itself, detect and track obstacles, and plot the optimal path of movement through its environment to complete its goal.

**Mapping**

In order to find a suitable path, a mobile robot has to have a concept of the location of objects surrounding it; a map. It is imperative that the map is updated accurately and constantly using accurate sensory data, as each other component of the movement planning algorithm relies on an accurate map to perform in a manner which does not result in what is known as catastrophic failure in engineering. Mapping itself has two main steps during standard operation, or the cycle which takes place outside of initialization. The first step is always gathering sensory input from the robot's surroundings.

In MINERVA, a museum tour guide robot designed by Thrun et al, (1999), two maps are generated in order to navigate the environment, an occupancy map and a ceiling texture map. Both maps are simultaneously refreshed using laser rangefinders, cameras, and odometry data. The ceiling texture map, a creative solution to the difficulties of localization in a dynamic filled with moving legs, utilizes a camera pointing towards the ceiling, a static environment, to get a somewhat reliable map. By using this ceiling texture map in parallel with the occupancy map, which uses laser rangefinders to map ground level obstacles, the researchers avoided the difficulty of having to identify and remove foreign objects from the later localization process, and additionally increases the accuracy of the map. However, this method of mapping can be atypical in modern robots. Wang, Thorpe, and Thrun (2003) provide an example of successful use of the current norm of area mapping in mobile robotics. The group created a system where the distance traveled and proximity of the surroundings are related, and a single map inclusive of dynamic values is generated. This single map doesn't attempt to rely on an entirely static information, rather just compensates for the constant fluctuation of the environment with a Markov machine learning system. Often times, employing this single map method will reduce the valuable computational resources needed to operate a robot.

As mentioned previously, recent research seems to favor the use of only one map, inclusive of moving objects is created, unlike the previous maps which filtered out moving objects in various ways. Robots running this singular map method must take into account these moving objects when mapping through a method known as DATMO, which stands for detection and tracking of moving objects. The mapping framework built in the mid 2000's, primarily around the Navlab8 and Navlab11 platform; a modified minivan and jeep respectively, is an excellent example of this single map approach. (Wang, Thorpe, Thrun, Hebert, & Durrant-Whyte, 2007) The developers used rangefinders and stereoscopic cameras to capture the environment of the robot indiscriminately, and then classified certain objects as moving based on their relative movement speed. This method is surprisingly successful, but it is not the most favorable solution for a real life environment, as the classification decreases in accuracy as the degrees of motion increase. Humans and other animals often have upwards of 200 degrees of motion. As one could imagine, if a self-driving car were to have a subpar ability to separate humans from other objects, it would be almost unable to function in the real world.

**Localization**

Localization, according to Dellaert, Fox, Burgard, and Thrun (2001), are the actions a robot takes to determine and track its location relative to a map using sensor data, which may be composed during or prior to the robot's interaction with its surroundings. Before current sensory data may be compounded to the correct position on the map, localization must occur. The initial stages of localization must be based on odometry data. Odometry data can be gathered in several ways, however, all methods share the major issue of noise. Readings from the sensor regularly fluctuate due to uncontrollable environment variables. This is also known as noise. Filtering this noise is necessary in localization because otherwise error builds up, effectively "snowballing". The rough distance between the actual location of the robot and the calculated location can be modelled by this system of equations:

$$c = \sum_{q-0}^{(n-1)} (d_q) + m,\ d_n = a - c$$

where ***n*** is the number of movement-data collection cycles, ***d*** is the distance from the actual location at cycle ***n***,***a***is the actual location, and ***c*** is the calculated distance.

**Filtering**

Because noise reigns supreme over any dataset generated by sensors, filtering is a problem that has been solved many different times in many different ways, with often widely varying degrees of success. Filtering inputs a raw, unprocessed signal from a sensor, and from there outputs a prediction of the accurate, true to life signal, that does not include noise. The noise is "filtered" out, thus giving the concept its name.

**Kalman's Filter.** One of the most commonly used filters in robotics and engineering is Rudolf E. Kalman's famous Kalman filter, and its relative, the Extended Kalman filter. Simon Levy (2016) covers the Kalman filter in depth while discussing his methods for accurately determining the elevation of an unmanned aerial system, outlining the filter as a rudimentary Bayesian machine learning system. The estimated difference between the actual value, which is noise exclusive, and the collected value, which is noise inclusive, is based on the previous prediction and the error margin. The estimated difference allows the algorithm to predict the current real, noise exclusive value based on the noise inclusive sensor reading to a degree of certainty. The error margin will often be generated using Gaussian distribution to determine what the most likely real value is. However, this method of filtering poses several problems to the roboticist who seeks to implement it in complex robots. As pointed out by Levy (2006) creating an error margin by comparing the predicted value to a Gaussian distribution poses several assumptions about the nature of noise, for example, that they will follow a Gaussian distribution, or any distribution for that matter, reliably. Additionally, Wang and associates (2007) used a Kalman filter for computer vision based object tracking, but found flaws in its heavy demand of computational resources, as the Kalman filter only models a single value at a time. Levy (2006) discusses his use of the Kalman filter in an UAV autopilot, which

**Monte Carlo Localization.** Alternatively in Mixture-MCL, a localization method introduced by Thrun, Fox, Burgard, and Dellaert (2000), particle filters, otherwise known as Sequential Monte Carlo methods, are used to address the numerous assumptions regarding noise made by the Kalman filter. Thrun (1999) outlines the particle filter as a system which invokes a large number of “particles”, each with its own heuristic weight. As explained by Teammco (2013), these weights are changed based the difference between the particle’s distance from a known location and the robot’s distance from this location. The particles with the lowest distance have their weights increased, and vice versa. Then, the particle can be moved in a random direction in order to determine whether that increases or decreases the weight. If a weight becomes too weak, it is eliminated and replaced with another one, with a large weight. The robot’s location can be derived from this cloud of heuristic values. This method of localization eliminates noise by removing particles that contribute error to the data set.

## Pathfinding

Once a map is generated and the robot is localized, the next piece of the movement planning puzzle is a pathfinding algorithm. Pathfinding itself is not a very difficult problem at its root, it actually can be solved in some cases by just plotting a line between two points, however, other solutions, like the A* algorithm, allow the same fundamental problem to be solved despite added challenges, like obstacles or dimension.

**A-Star.**The A*, or A-Star pathfinding algorithm is an extremely popular pathfinding algorithm used in everything from video games to homemade missile guidance systems. According to Cui and Shi (2011); A-Star functions with the use of nodes, or locations, which can have several states, open and closed. These nodes in robotics and mapping can be defined as a physical location of a certain size. A* uses recursion to select a square and determine whether it is obstacle. If a node is not an obstacle, it is given the closed state, meaning it is a node that can be moved past. Nodes which are unexplored have an open state, and those which are obstacles have neither states, so they are not considered at all. The algorithm checks every single combination and picks out the path which meets certain criteria, like the least distance or fewest turns and curves in the path. This algorithm has been utilized wildly in video games due to its computational and intrinsic simplicity, allowing multiple to be run in parallel for different game "AI's".

**Genetic Algorithm Hybrid.** While the A-Star algorithm is popular, other approaches are often taken as well in order to gain certain forms of paths. An emerging method of path finding is a hybridization of a genetic algorithm and another pathfinding algorithm. A genetic algorithm is fairly self-explanatory; in that it features "natural" selection. Different results from different variations of an algorithm created during initialization are pitted against each other and selected for fitness. The large sample of generated algorithms allow for the most efficient result to be picked for a situation, for example, the path with the smoothest curves, so as to minimize the possibility of the robot flipping or breaking during a sharp turn, or being unable to accurately follow the path due to the sharpness of turns. This pathfinding method has even been employed in UAV navigation systems. (Nikolos, Valavanis, Tsourveloudis, & Kostaras, 2003) The group first generates a random set of trajectories from the current position to an end point through three dimensional space. After this, algorithms are selected for "smoothness" and integrity. Those which are smooth and are free of collisions "reproduce" to ensure those traits persist throughout the rest of the new synthesized algorithms. Those which lack those traits are not selected and thus die out, and the new algorithms lack the “losing” traits.

## Hardware

Hardware in this context is defined as a set of physical effectors with which the control software and observes interacts with its physical environment. Without a mechanical body capable of traversing through the terrain plotted by the path-finding algorithm, the mobile system is useless. Using this line of logic, the designer who maximizes the versatility and success rate of the locomotion system is thus maximizing the robot’s own ability to complete tasks assigned to it. (Apostolpoulos, 2001)

### Actuator Control

Actuator control can be defined as the pattern of feedback and stimuli provided to the actuator in order to control it with accuracy. A common example of this in robotics would be a proportional integral derivative controller. These controllers utilize a loop of correction between the actual point of an actuator’s end effector and the commanded, or set. These can be found in practically every robot, but certainly every robot that must localize, as they provide accountability and tracking needed to accurately track distance traveled by revolutions of the actuator. We can assume that MINERVA, the previously discussed robot for giving tours at a museum, used PID to control the motors that carried it around the museum, however, it was not mentioned directly. The concept is a relatively general subject in engineering, and doesn’t need an in depth review.

**Locomotion**

Locomotion systems can be first be subdivided into two groups based on suspension, passive suspension and active suspension. The active suspension system utilizes a single or groups of actuators to change the angles of appendage angles or the format of the drive structure in relation to the terrain while moving over it. The passive suspension system simply conforms to the terrain, passively, as the name suggests, allowing a robot to move smoothly and fluidly over most small obstacles. (Thueer, Krebs, and Siegwart, 2006) However, under-actuation can be harder to model mechanically. Pedal robots are excellent candidates for active suspension as appendage segments must be moved differently according to acceleration, incline, and direction, and the direct control of torque or speed at specific joints without requiring the change of another joint's state can be useful in implementing new or unplanned maneuvers in the robot, despite decreased passive handling of terrain.

**Replication of Biological Systems**

Nature has provided inspiration for scientists over the centuries, and that most likely will not change soon. Recently, spurred by the advancement of components such as transistors and actuators, robotics has taken notice of nature's wisdom. Robots like MIT's Cheetah, Boston Dynamics BigDog, and ETH Zurich's StarLETH mimic quadrupedal morphology. A US Government funded study found about half of the world's landmass to be un-traversable by conventional wheeled vehicle. (Bartlett, Belsdorf, Deutschman, & Smith, 1969) This new class of robots has proven themselves to be extremely capable of moving across rough and difficult terrain, filling that gap.

**The dog.** The anatomy and morphology of a dog is an excellent candidate for replication due to its demonstrated ability to live alongside humans almost wherever they may be. The dog is easily able to move across rough terrain and has proved its ability to move reliably in urban, suburban, and rural areas through its prevalence in our lives. However, fully modelling the over 200 degrees of freedom available to a dog is currently practically impossible, although attempts at a reasonable reproduction can be made. After a certain point, the extra dexterity afforded by a dog's limbs would be unnecessary, sort of redundant to a robot, depending on the task it was meant to undertake. Despite limitations to the full realization of its anatomy, the dog has been widely modelled in biomimetic robotics, all with relatively similar results, and research has been conducted to analyze the dog's musculoskeletal system in order to learn more about how the system can be copied. In fact, Biewiener, a prolific roboticist, studied the locomotion of both dogs and goats to determine exactly how it should be recreated in BigDog, the Boston Dynamics robot previously mentioned, with focus on leg spring force and bone structure. (Biewiener, 1983) (Lee, Biewiener, 2011)

**The goat.** Goats have received similar attention from the robotics community for their incredible agility. Although no majorly successfully goat mimicking robot has been constructed to date, the process of creating one would be similar to dog-based robots.

## Sensors

Sensors are key to any form of robotics. They allow the robot to interact with and understand their environment, including all of the physical obstructions to the robot's path or other intricacies of the surrounding area. In this form of robotics application, the autonomous navigating mobile robot, the robot must have a rather large number of sensors in order to be able to take into account every variable needed during the course of action. In mobile robots this sensory array can include IMU's, which is an integrated accelerometer and gyroscope sensor suite, rangefinders of the laser, ultrasonic, or IR variants, and various forms of variable resistors often repurposed into force sensors. These allow the robot to detect and avoid obstacles, stay upright, move at specified angles or adapt to inclines, and shift its center of mass based whenever the command to turn is called within the robot.

**Range finding sensors.** Range finding sensors are essential to an autonomous mobile robot's navigational success by providing the robot with the distance between the robot and the nearest object in the sensor's scope. There are four main types of range finding sensors, ultrasonic-based, IR-based, laser-based, and imaging-based, each with their own benefits and drawbacks. However, there are certainly more than just these methods of determining distances, the aforementioned selection are just the most available to the researcher in this circumstance.

***Ultrasonic range finding (SONAR).*** Ultrasonic range-finding operates off of the time-of-flight mechanic found in most range-finding sensors. (Carullo and Parvis, 2001) First, an ultrasonic pulse is emitted from a speaker. Once the pulse collides with a structure, it will bounce back towards the microphone. The sensor records the time it took for the sound to travel back to complete the cycle, divides it by two, and multiplies that by the approximate speed of sound, thus getting the distance travelled. However, sound waves diffuse as they travel. This means as the distance between emission and structure increases, the accuracy decreases, as the sensor is more prone to picking up noise, or obstacles not directly in front of the sensor. These sensors were used on the previously discussed MINERVA robot.

***IR range finding.*** Analog infrared range finding works almost completely differently than ultrasonic methods. Analog infrared range finding instead uses the central concept that the light strikes the receiver to calculate the distance, with smaller angles being a shorter distance. These sensors are often very inaccurate due to uncontrollable environmental variables' effect on the angle perceived by the sensor.

***Computer vision range finding.*** Computer vision range finding comes in many different variants, there is no uniform way to find the depth, as it is often called, with imagery. Analysis of images for certain patterns of pixels in the form of lines, gradients of shading, or other conditions will allow for the perception of certain distances or depths. (Jarvis, 1983) Computer vision was used successfully to detect certain features and assist in localization in the MINERVA robot discussed previously.

Methodology Regarding the Creation of an Autonomous Quadrupedal Robot

## Participants/Subjects

The subject of this study will be a single, biomimetic, quadrupedal robot designed and built by the researcher. It will be created specifically for the purpose of the study.

## Apparatus/Measures

This study will contain a single main apparatus to aide measurement; the isolated testing environment. The apparatus will consist of a small, walled in area with various obstacles and terrain formations, such as patches of different types of soil and slight inclines, as well as a preset destination that the subject should navigate to. The primary purpose of this device is to provide both surroundings and external stimuli for the subject to react and respond to, as well as a goal for the subject to achieve.

The subject's performance in the testing environment will be graded on a simple scale developed by the researcher. The subject will be given a goal of navigating from its initial location to a known end location, similar to Pomerleau's ALVINN self-driving car prototype (1989). The three point scale will measure the subject's success in completing the goal. If the robot successfully moves to the end point without incurring any damage to its ability to operate, it gets a one. If the subject moves to the destination but does undergo damage that would threaten future operation, meaning it has a local failure, the robot receives a zero. If the subject does not reach its destination it gets a negative one score on the scale. Of course, both the subject and the testing apparatus will require materials to construct. The subject will require far more, however, than the testing apparatus. The subject will need micro-LIDAR sensors, 14 servo motors, 3 to actuate each leg, one for the spine, and one for the head, a IMU with a built in magnetometer for use as a compass, a motor controller and power distribution board as well as an onboard microprocessor capable of communicating data to the server for processing.

The apparatus will require wood for its outer boundaries, soda cans or other easily placed objects for obstacles, some rolling obstacles, like tennis balls, some sand or loose soil as well as gravel to mimic different terrains and form small hills, and some colored tape to mark the destination and initial location of the subject.

## Procedures

Prior to the data collection, both the subject and the testing apparatus must be constructed. The physical construction phase will last the longest, with the assembly of the subject taking upwards of eight hours once all the components are available. The programming phase will be shorter, with a good time estimate being around four hours of coding, and potentially more for loading and code onto the subject and the server and bug-testing. Next, the testing environment must be created. Collection of materials and general construction should take no more than two hours, due to the extreme simplicity of the environment. After both objects are constructed, the subject can be tested in the experimentation apparatus, unlike most other research done on the development of robotics, where the subject is analyzed rather than its performance. (Balaguer, Giménez, Pastor, Padron, & Abderrahim, 2000) The subject will be given a destination’s distance and cardinal location in comparison to its initial location and be set into autonomous mode to navigate the environment. A fifteen minute time limit will be in place to ensure that the true nature of the subject’s performance in that run is accurately represented in the data. After the fifteen minute time limit has passed the scale will be applied to the subject’s performance. Then the movable obstacles inside of the testing area will be re-arranged to determine if the subject's performance is consistent, and a new destination will be selected and the same information will once again be provided to the subject. This cycle of randomization and testing will be repeated no fewer than 20 times to get an adequate body of data. The single value from each test run will be entered into a scatter plot for visualization of the grouping of the data.

## Evaluation

Once the data is collected, it can be evaluated to determine the capability of the subject to navigate to a chosen destination in a new environment. If the average is at or above a zero, the robot will have proven to be capable of generally being capable of autonomously circumventing obstructions in novel environments. Because a random yet realistic scenario was generated for each individual test, the measure of adaptability, the desired trait, will come from consistent success. If the subject scores a non-negative average, the researcher will understand that the subject's development was a success.

## Conclusion

If the subject fails to score a non-negative the study will have failed to design a successful subject, as the data proves that the subject would certainly be unsuccessful in the real, more chaotic world. The researcher believes that the study will confirm the view that biomimetic design strategies and adaptable design will triumph in their versatility.

**References**

Henderson, H. (2006). *Modern robotics: building versatile machines*. Infobase Publishing.

Hutter, M., Gehring, C., Bloesch, M., Hoepflinger, M. A., Remy, C. D., & Siegwart, R. O. L. A. N. D. (2012). StarlETH: A compliant quadrupedal robot for fast, efficient, and versatile locomotion. In *15th International Conference on Climbing and Walking Robot-CLAWAR 2012* (No. EPFL-CONF-181042).

Pfeifer, R., Lungarella, M., & Iida, F. (2007). Self-organization, embodiment, and biologically inspired

robotics. *science*, *318*(5853), 1088-1093.

Raibert, M., Blankespoor, K., Nelson, G., Playter, R., & Team, T. B. (2008, July). Bigdog, the rough-terrain quadruped robot. In Proceedings of the 17th world congress (Vol. 17, No. 1, pp. 10822-10825). Proceedings Seoul, Korea.

Robotics: A Brief History. (n.d.). Retrieved March 21, 2017, from https://cs.stanford.edu/people/eroberts/courses/soco/projects/1998-99/robotics/history.html

Summary. (n.d.). Retrieved May 15, 2017, from https://www.bls.gov/ooh/transportation-and-material-moving/taxi-drivers-and-chauf Feurs.htm

Urmson, C. (2008). Self-driving cars and the urban challenge. *IEEE Intelligent Systems*, *23*(2).

Yamauchi, B. M. (2004, September). PackBot: A versatile platform for military robotics. In *Defense and Security* (pp. 228-237). International Society for Optics and Photonics.

Yim, M., Shen, W. M., Salemi, B., Rus, D., Moll, M., Lipson, H., ... & Chirikjian, G. S. (2007). Modular self-reconfigurable robot systems [grand challenges of robotics]. *IEEE Robotics & Automation Magazine*, *14*(1), 43-52.

Apostolopoulos, D. (2001). Analytic Configuration of Wheeled Robotic Locomotion. Research Showcase @ CMU

Bartlett, G. E., Belsdorf, M. R., Deutschman, J. N., & Smith, R. L. (1969). On the prediction of off-
road vehicle system mobility (No. 690150). SAE Technical Paper.

Biewiener, A. A. (1983) Alllometry of Quadrupedal Locomotion: The Scaling of Duty Factor, Bone, Curvature, and Limb Orientation to Body Size. J. exp. Bio,105, 147-171. Retrieved March 9, 2017, from http://biomimetic.pbworks.com/f/Allometry%20of%20quadrupedal%20locomotion%20theBiewener.pdf

Carullo, A., & Parvis, M. (2001). An ultrasonic sensor for distance measurement in automotive applications. IEEE Sensors Journal, 1(2), 143. doi:10.1109/jsen.2001.936931

Cui, X., & Shi, H. (2011). A*-based Pathfinding in Modern Computer Games . IJCSNS International Journal of Computer Science and Network Security,11(1), 125-130. Retrieved March 9, 2017, from

Dellaert, F., Fox, D., Burgard, W., & Thrun, S. (1999). Monte Carlo Localization for Mobile Robots. Computer Science Department, Carnegie Mellon University. Retrieved February 24, 2017.

Evans, H. E., & De Lahunta, A. (2013). Mille's Anatomy of the Dog. Amsterdam, Netherlands: Elsevier Health Sciences.

Guo, Y., & Parker, L. (2002). A distributed and optimal motion planning approach for multiple mobile robots. Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292). doi:10.1109/robot.2002.1013625

Jarvis, R. A. (1983). A perspective on range finding techniques for computer vision. IEEE Transactions on Pattern Analysis and Machine Intelligence, (2), 122-139.

Levy, S. D. (2013). Kalman Tutorial. Retrieved March 10, 2017, from https://home.wlu.edu/~levys/kalman_tutorial/kalman.pdf

Lee, D. V., & Biewener, A. A. (2011). BigDog-inspired studies in the locomotion of goats and dogs. Integrative and comparative biology, icr061.

Nikolos, I. K., Valavanis, K. P., Tsourveloudis, N. C., & Kostaras, A. N. (2003). Evolutionary algorithm based offline/online path planner for UAV navigation. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 33(6), 898-912.

Thrun, S. (2002). Particle Filters in Robotics. In Proceedings of Uncertainty in AI. Retrieved March 10, 2017, from http://robots.stanford.edu/papers/thrun.pf-in-robotics-uai02.pdf

Thrun S., Bennewitz M., Burgard W., Cremers A. B., Dellaert F., Fox D., H¨ahnel D., Rosenberg C., Roy N., Shulte J., & Schulz J. (1999) *MINERVA: A Second-Generation Museum Tour-Guide Robot*

Thrun, S., Fox, D., Burgard, W., & Dellaert, F. (2001). Robust Monte Carlo localization for mobile robots. Artificial Intelligence, 128(1-2), 99-141. doi:10.1016/s0004-3702(01)00069-8

Thueer, T., Krebs, A., & Siegwart, R. (2006). Comprehensive Locomotion Performance Evaluation of All-Terrain Robots. 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. doi:10.1109/iros.2006.281954

Wang, C. C., Thorpe, C., Thrun, S., Hebert, M., & Durrant-Whyte, H. (2007). Simultaneous localization, mapping and moving object tracking. The International Journal of Robotics Research, 26(9), 889-916.

Wang, C. C., Thorpe, C., & Thrun, S. (2003, September). Online simultaneous localization and mapping with detection and tracking of moving objects: Theory and results from a ground vehicle in crowded urban areas. In Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on (Vol. 1, pp. 842-849). IEEE.

Balaguer, C., Giménez, A., Pastor, J. M., Padron, V. M., & Abderrahim, M. (2000). A climbing autonomous robot for inspection applications in 3d complex environments. Robotica, 18(03), 287-297.

Pomerleau, D. A. (1989). ALVINN, an autonomous land vehicle in a neural network (No. AIP-77). Carnegie Mellon University, Computer Science Department.