A Soft Landing Model and an Experimental Platform as an Introductory Control Design Tool[1]

Hakan Yasarcan and Togay Tanyolaç
Industrial Engineering Department
Bogazici University
Bebek – Istanbul 34342 – Turkey
hakan.yasarcan@boun.edu.tr; caloynat@gmail.com

Abstract
This paper presents a soft landing model and an experimental platform. The aim of the modeling effort is to transparently represent the process of landing a spacecraft on the surface of a celestial body. The process of landing is an interesting problem because there are two main contradictory performance criteria to be met simultaneously; the landing duration should be as short as possible, but at the same time crashing the spacecraft to the surface should be avoided. If the only criterion was to prevent crashing the spacecraft, that would not be difficult to achieve by slowing down the landing process. However, long landing duration necessitates extensive use of fuel, which should also be avoided. As a summary, the main goal in the soft landing problem is to land the spacecraft as gently and as fast as possible. Many real life complexities such as delays caused by actuators and measurement processes are not represented in the model. Even under the simplifying model assumptions, the main goal of the soft landing problem still remains a challenging one because the two state variables "height" and "velocity" can only be indirectly controlled. The model and the modeling process presented in this paper will serve as a valid model construction case to be used in teaching. We also developed a platform for simulation experiments. Our simulation-based discovery learning environment/platform can be used as an introductory control design tool for physics, engineering, and interested social sciences students. The model and the platform can also be used to introduce dynamic complexity.

Keywords: soft landing; model transparency; modeling process; valid model construction; experimental platform; discovery learning.

[1] *This research is supported by a Marie Curie International Reintegration Grant within the 7th European Community Framework Programme (grant agreement number: PIRG07-GA-2010-268272) and also by Bogazici University Research Fund (grant no: 5025-10A03P9).*

1. Introduction

Soft landing is an interesting and challenging problem in space exploration. The process of landing is a challenging task because there are two main contradictory performance criteria to be met simultaneously; the landing duration should be as short as possible, but at the same time crashing the spacecraft to the surface should be avoided. In order to achieve a fast and safe landing on the surface of a celestial body, the landing process should be controlled. When landing on celestial bodies with no atmosphere (e.g. the moon), deceleration strategies that rely on the drag force (e.g. a parachute) do not work due to the absence of atmospheric molecules. Therefore, a reverse force thruster, which will decelerate the vehicle, is needed (see Figure 1). At the instant of landing, an impact force is generated depending on the mass, velocity, and the landing gear specifications of the spacecraft. For a successful landing, this impact force must be under a certain limit and, ideally, it should be as low as possible so as not to harm the vehicle. We assumed a constant mass and fixed specifications for the landing gear. Thus, the magnitude of the impact force can only be controlled via controlling the velocity, which should be within certain limits to prevent a crash. If the only criterion was to prevent crashing the spacecraft, that would not be difficult to achieve by slowing down the landing process. However, long landing duration necessitates extensive use of fuel, which should also be avoided. Therefore, another goal in landing is to decrease the time to land. Consequently, a reasonable landing occurs when the vehicle descends to the surface quickly, but decelerates safely to low velocity values before the instant of landing (Liu, Duan, and Teo, 2008; Zhou et al., 2009).

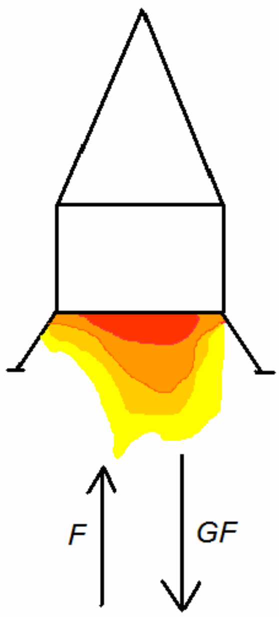

Figure 1: Free body diagram of the vehicle with a control force (F) generated by the reverse force thruster and the gravitational force (GF)

We modeled the soft landing challenge using System Dynamics (SD) simulation methodology (Barlas, 2002; Forrester, 1961 and 1971; Sterman, 2000). SD has a strong focus on the correct representation of the problem related elements of a system, which increases the validity of the constructed simulation model. Accordingly, all model variables and parameters were carefully selected and added to our soft landing model; they all are related to the soft landing problem and in accordance with the theory of motion. As a result, physical complexities that do not significantly contribute to the dynamics of the soft landing problem are ignored. We focused our modeling efforts only on the vertical movement and completely ignored movement in the horizontal axes because we believe that simplicity is a key element in increasing comprehension of the structure of and the dynamics generated by the model. Thus, we paid special attention in including/excluding variables and parameters to and from the model (Hübler 2007; Saysel and Barlas, 2006; Yasarcan, 2010). There are no hidden variables or parameters in the model such as numerical values in equations; we explicitly represented all of them with their corresponding units and verified the dimensional consistency of our model. Before finalizing the model, we carried out many simulation experiments, carefully examined the generated dynamics, and made necessary improvements to the model by correcting the structure and calibrating the parameter values.

The aim of the modeling effort was to transparently represent the process of landing a spacecraft on the surface of a celestial body. Explicit representation of the model variables and parameters serves this purpose. We hope that the level of transparency we achieved and SD representation tools (i.e. stock flow diagram; causal-loop diagram) will facilitate sharing and understanding of the model structure and dynamics, as one of the aims is to use the model in teaching. We also developed an experimental simulation platform based on the model that will further enhance the model's value as a learning tool. The model and its assumptions will be presented in the next section. In the latter sections, we will introduce the simulation-based discovery learning environment (i.e. experimental simulation platform) and discuss dynamics.

2. The Model Structure and Equations

In this study, we first constructed a stock-flow model of the soft-landing problem, which is given in Figure 2. This diagram represents only the physical structure of the problem described in the previous section; it does not represent the controller (e.g. a human decision maker, a computer). *Height* (i.e. the vertical distance between the spacecraft and landing surface) and *Velocity* (i.e. the vertical velocity) are the two stock variables (accumulations, system state variables) in the model, which are represented as boxes (see Figure 2). The stock equations 2 and 4 are approximate integral equations. DT (simulation time step) in these equations is set to 2^{-8} (1/256) seconds, which is sufficiently small in emulating continuous time behavior. *Velocity*, which is a stock variable, is at the same time the one and only flow of *Height*. *Velocity* has a single flow too; *Acceleration*. In our model diagram (Figure 2) there are only two flows, which are represented by thick arrows with a valve in the middle. Flows define the rate that stocks change. Hence, *Height* is controlled via *Velocity*, *Velocity* via *Acceleration* (equations 2 and 4). We roughly selected the initial conditions for the spacecraft so as to observe important dynamics that the model can generate (equations 1 and 3). For example, if *Height* was set to a very low initial value, it would not be possible to observe how the vehicle behaves before it enters the very final stage of landing.

$$Height_0 = 1000 \quad [m] \tag{1}$$

$$Height_{t+DT} = Height_t + Velocity_t \cdot DT \quad [m] \tag{2}$$

$$Velocity_0 = -10 \quad [m/s] \tag{3}$$

$$Velocity_{t+DT} = Velocity_t + Acceleration_t \cdot DT \quad [m/s] \tag{4}$$

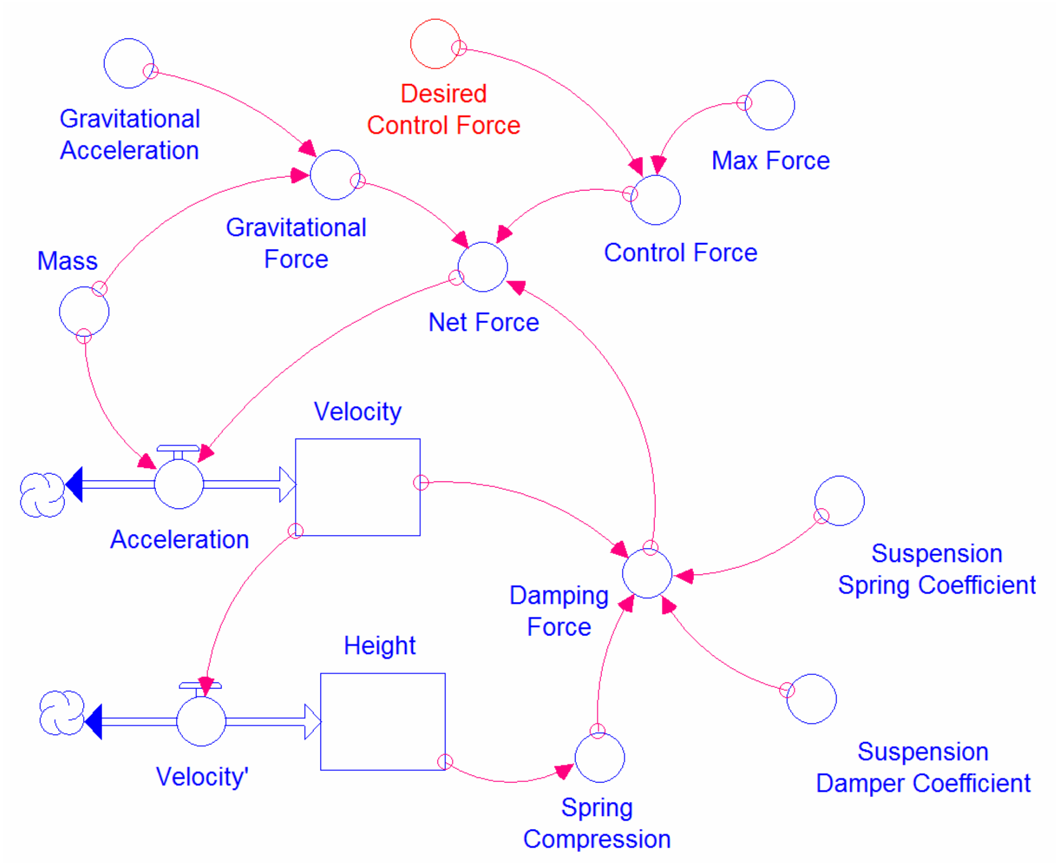

Figure 2: Stock-flow diagram of the model

The thin arrows in Figure 2 represent causal functional relations that define the non-stock variables. Accordingly, *Net Force* and *Mass* determine *Acceleration* (Equation 5). In our model, *Mass* is a constant because we ignored the change in the mass due to fuel consumption (Equation 6). By doing so, we kept the model fairly simple to avoid an extra load of information that would complicate the essential understanding of the structure of the model.

$$Acceleration = Net\ Force / Mass \quad \left[m / s^2 \right] \tag{5}$$

$$Mass = 1000 \left[kg \right] \tag{6}$$

$$Net\ Force = Gravitational\ Force + Damping\ Force + Control\ Force \quad \left[N \right] \tag{7}$$

Height is controlled via *Velocity* (Equation 2), *Velocity* via *Acceleration* (Equation 4), *Acceleration* via *Net Force* (Equation 5), and *Net Force* via *Control Force* (Equation 7)[2]. The control feedback loop structure also includes the controller, which determines *Control Force* via *Desired Control Force*. The natural inputs to the controller are *Height* and *Velocity*. A simplified causal loop diagram showing these relations and two negative (counteracting) feedback loops within the control feedback loop structure can be seen in Figure 3. Although, every control system involves delays in measuring/perceiving actual conditions (Yasarcan, 2011), we ignored such delays in our model for the sake of simplicity and assumed that the controller has instantaneous access to the current values of *Height* and *Velocity*. We also ignored delays caused by actuators. Explicitly modeling delays caused by actuators and measurement processes increases the model complexity (Atay, 2009; Barlas, 2002; Forrester, 1961 and 1971; Michiels and Niculescu, 2007; Sterman, 2000; Yasarcan, 2010 and 2011; Yasarcan and Barlas, 2005; Zhang, Park, and Chong, 2009).

Positive *Height*, *Velocity*, *Acceleration*, and force directions are upward from the surface. *Height* equals zero means that the vehicle touches the ground, but the springs of the landing gear are at rest, so they bear no force at *Height* equals zero. Thus, when the vehicle comes to a static equilibrium, the springs of the landing gear get compressed balancing the weight (*Gravitational Force*) of the vehicle and *Height* becomes slightly less than zero. See the assumption regarding the *Suspension Spring Coefficient* at the end of this section.

Gravitational Force, *Damping Force*, and *Control Force* add up to the *Net Force* acting on the vehicle (Equation 7). *Gravitational Force* acts on the vehicle due to mass

---

[2] One Newton amounts to the force needed to increase the velocity of a one kilogram body of mass by one meter per second in one second ($N = kg \cdot m / s^2$).

and gravity (Equation 8). *Gravitational Acceleration* is assumed to be constant during landing; in the model, it does not change with the distance to the surface (Equation 9). Corollary to constant *Mass* (Equation 6) and constant *Gravitational Acceleration* (Equation 9), *Gravitational Force* is also a constant (Equation 8).

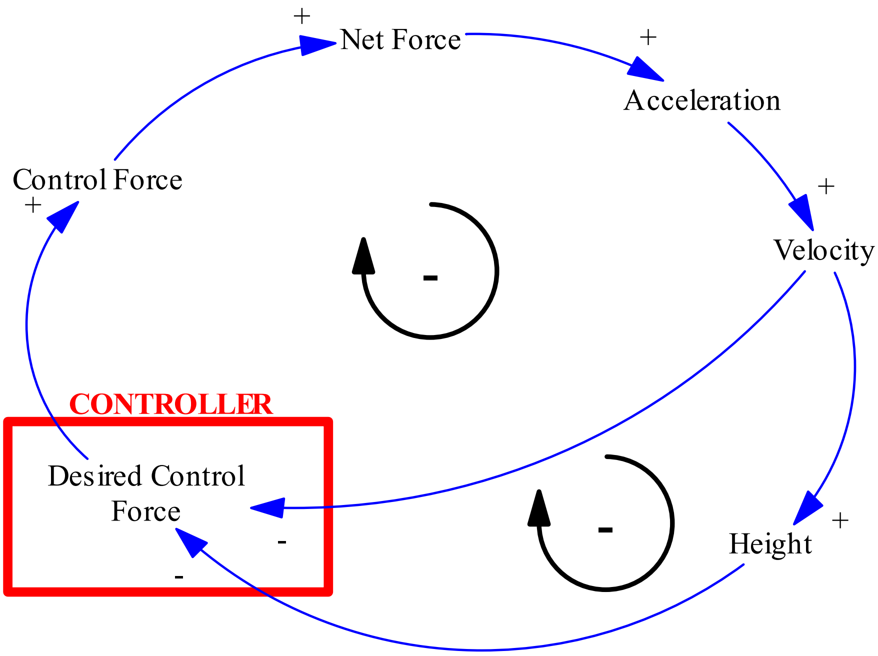

Figure 3: Causal-loop diagram of the control feedback loop structure

$$Gravitational\ Force = Mass \cdot Gravitational\ Acceleration = 8{,}870 \quad [N] \tag{8}$$

$$Gravitational\ Acceleration = 8.87 \ [m/s^2] \tag{9}$$

The gravitational acceleration of the celestial body to be landed on is assumed to be equal to the surface gravitational acceleration of Venus that is 8.87 m/s^2 (Equation 9). Note that the assumed landing conditions other than the gravitational acceleration do not resemble the conditions of Venus at all. Venus has a thick atmosphere, but we aimed to capture the difficulty caused by the absence of drag. Hence, we assumed zero drag force.

The landing gear of the spacecraft is comprised of dampers and springs. *Damping Force*, which is a result of the compression of the landing gear, is generated after the spacecraft contacts the landing surface (Equation 10). To be able to correctly represent the conditional existence of *Damping Force*, we also defined a variable named *Spring*

Compression, which represents the amount of compression of the landing gear (Equation 11).

$$Damping\ Force = \left\{ \begin{matrix} 0, & Spring\ Compression = 0 \\ \begin{pmatrix} Suspension \\ Spring \\ Coefficient \end{pmatrix} \cdot \begin{pmatrix} Spring \\ Compression \end{pmatrix} - \begin{pmatrix} Suspension \\ Damper \\ Coefficient \end{pmatrix} \cdot Velocity, & \text{otherwise} \end{matrix} \right\} [N] \quad (10)$$

$$Spring\ Compression = \begin{pmatrix} 0, & Height \geq 0 \\ -Height, & \text{otherwise} \end{pmatrix} \quad [m] \quad (11)$$

The equilibrium level for *Spring Compression* is a constant named *Landing Gear Rest Compression*, which is the amount of the compression in the springs caused solely by the weight of the spacecraft on the target celestial body. *Landing Gear Rest Compression* is arbitrarily selected to be 0.5 meters (Equation 12) and the value of *Suspension Spring Coefficient* (Equation 13) is selected such that the desired *Landing Gear Rest Compression* is achieved after the touchdown (i.e. *Height* asymptotically approaches to its equilibrium value after the touchdown, which is equal to minus one times *Landing Gear Rest Compression*). *Suspension Damping Factor* determines the dynamics after the touchdown (Equation 14). When it is less than 2, it gives underdamped behavior; when it is more than 2, it gives overdamped behavior; when it is equal to 2, it gives critically damped behavior (Åström and Murray; 2008). Finally, *Suspension Damper Coefficient* is calculated based on the other landing gear parameter values (Equation 15).

$$Landing\ Gear\ Rest\ Compression = 0.5 \quad [\text{m}] \quad (12)$$

$$Suspension\ Spring\ Coefficient = \frac{Gravitational\ Force}{Landing\ Gear\ Rest\ Compression} = 17{,}740 \quad [N/m] \quad (13)$$

$$Suspension\ Damping\ Factor = 2 \quad [\text{dimensionless}] \quad (14)$$

$$\begin{pmatrix} Suspension \\ Damper \\ Coefficient \end{pmatrix} = \begin{pmatrix} Suspension \\ Damping\ Factor \end{pmatrix} \cdot \sqrt{\begin{pmatrix} Suspension \\ Spring\ Coefficient \end{pmatrix} \cdot Mass} = 8{,}424 \quad \left[\frac{N \cdot s}{m}\right] \quad (15)$$

Desired Control Force determined by the controller, which is explained in the fourth section, is an input to *Control Force* of the reverse force thruster. *Control Force* cannot be more than the maximum force applicable by the thruster (equations 16 and 17).

$$\text{Control Force} = \begin{cases} \text{Desired Control Force}, & \text{Desired Control Force} \le \text{Max Force} \\ \text{Max Force}, & \text{otherwise} \end{cases} \quad [N] \tag{16}$$

$$\text{Max Force} = 30{,}000\ [N] \tag{17}$$

There are no hidden variables or parameters in the model. For example, the summation of *Gravitational Force*, *Damping Force*, and *Control Force* could be used in the *Acceleration* formulation instead of *Net Force* (equations 5 and 7). However, we chose to explicitly represent *Net Force* in order to increase understanding of the model. We could also embed the numerical values of the parameters directly in the formulations of the variables. However, the transparency of the model significantly increases by explicitly naming the parameters and providing their units, which also facilitates model sharing. Furthermore, the explicit representation of variables, parameters, and their corresponding units smoothes the process of verification and validation and contributes to the validity of the model. As a part of the study, we also carried out many simulation experiments, carefully examined the generated dynamics, and made necessary improvements to the model by correcting the structure and calibrating the parameter values, which is not presented in this paper so as not to make it unnecessarily long. After this meticulous modeling process, a reasonably valid representation of the landing process is achieved. We intentionally left out many real life complexities in order to keep the model simple and increase the comprehension of the structure of and the dynamics generated by the model. As a result, physical complexities that do not significantly contribute to the dynamics of the soft landing problem are ignored. Accordingly, we carefully include/exclude variables and parameters to and from the model.

The simplifying model assumptions are given below:

- The movement of the spacecraft in the horizontal axes is not modeled. Spacecraft is assumed to move only vertically.
- There is no atmosphere in the landing area, thus no air friction exists that would cause a drag force on the vehicle (Equation 7).
- *Gravitational Acceleration* is assumed to be constant during landing, it does not change with the distance to the surface (Equation 9).
- *Mass* is a constant, the change in the mass due to fuel consumption is ignored (Equation 6).
- The landing gear has fixed specifications, *Suspension Spring Coefficient* and *Suspension Damper Coefficient* are both constants (see equations 12-15).

- *Suspension Spring Coefficient* is selected so that the equilibrium value for *Spring Compression* is 0.5 meters (i.e. the equilibrium value for *Height* is -0.5 meters) (see Equation 12).
- *Suspension Damper Coefficient* is selected so that critically damped behavior is obtained after the touchdown (i.e. after the touchdown, the vehicle asymptotically approaches to its equilibrium height value) (Equation 14).
- There are no delays caused by actuators; *Desired Control Force* generated by the controller affects *Control Force* without a time lag (Equation 16).
- Information flow from the system to the controller is perfect and instantaneous; There are no errors or delays caused by measurement processes (Equation 18).

3. Dynamic Behavior of Landing

As described in the previous section, *Height* is controlled via *Velocity* (Equation 2), *Velocity* via *Acceleration* (Equation 4), *Acceleration* via *Net Force* (Equation 5), and *Net Force* via *Control Force* (Equation 7). The control feedback loop structure also includes the controller, which determines *Control Force* applied by the reverse force thruster via *Desired Control Force* (Figure 3). In order to obtain a reasonable value for *Desired Control Force*, the controller should consider the system state variables (i.e. *Height* and *Velocity*). Only by doing so is it possible to reach the aim of landing the spacecraft as gently and as fast as possible. Recall that we intentionally left out many real life complexities in order to keep the model simple. Even under simplifying assumptions, the control task still is not a straightforward one. Despite the fact that one of our simplifying assumptions is that the values of the state variables *Height* and *Velocity* are measured/perceived instantaneously and without error, it is necessary to develop a proper control heuristic[3]. The main reason for the difficulty is that the control task requires simultaneous control of *Height* and *Velocity*, which –due to the physical structure of the problem– can only be indirectly affected by the reverse force thruster; *Height* and *Velocity* have inertia; their values do not change instantaneously (see Figures 1 and 2 and equations 1-7). The addition of delays caused by actuators to the model would further complicate the control task by amplifying the effect of the modeled inertia[4].

---

[3] For example, see Yasarcan (2011) for the significance of and difficulties introduced by measurement/perception delays.

[4] For example, see Yasarcan and Barlas (2005) for different types of delays between *Desired Control Force* and *Control Force* (i.e. delay between “control flow” and “acquisition flow”, and delay between “desired control flow” and “control flow”), and the effects of these delays.

The stock-flow model given in Figure 2 and defined by equations 1-17 describes the structure of the soft landing problem excluding the controller (see Figure 3). The equations of the heuristic suggested for the controller are given below:

$$\begin{aligned}&\mathit{Desired\ Net\ Force} = \\ &\quad -\begin{pmatrix}\mathit{Velocity}\\ \mathit{Coefficien\,t}\end{pmatrix}\cdot \mathit{Velocity} + \begin{pmatrix}\mathit{Height}\\ \mathit{Coefficien\,t}\end{pmatrix}\cdot\left(\mathit{Target\ Height} - \mathit{Height}\right)\quad [N]\end{aligned} \tag{18}$$

$$\mathit{Velocity\ Coefficient} = 200 \quad [N\cdot s/m] \tag{19}$$

$$\mathit{Height\ Coefficient} = 10 \quad [N/m] \tag{20}$$

$$\mathit{Target\ Height} = -\mathit{Landing\ Gear\ rest\ Compression} \quad [m] \tag{21}$$

$$\begin{pmatrix}\mathit{Desired}\\ \mathit{Control}\\ \mathit{Force}\end{pmatrix} = \begin{Bmatrix}\begin{pmatrix}\mathit{Desired}\\ \mathit{Net\ Force}\end{pmatrix} + \begin{pmatrix}\mathit{Gravitatio\,nal}\\ \mathit{Force}\end{pmatrix}, & \mathit{Landing\ \ State} = 0\\ 0, & \mathit{Landing\ \ State} = 1\end{Bmatrix}\quad [N] \tag{22}$$

$$\mathit{Landing\ State}_0 = 0 \quad [\text{dimensionl ess}] \tag{23}$$

$$\begin{aligned}&\mathit{Landing\ State}_{t+DT} = \\ &\quad \mathit{Landing\ State}_t + \begin{pmatrix}1, & \mathit{Landing\ \ State}_t = 0,\ \mathit{Height}_t < 0\\ 0, & \text{otherwise}\end{pmatrix}\quad [\text{dimensionl ess}]\end{aligned} \tag{24}$$

The core of the heuristic is given by Equation 18. It simply tries to achieve a balance between slowing down the vehicle and trying to close the distance between the vehicle and the surface of the celestial body. For details see Tanyolaç and Yasarcan (2012). Equations 22-24 makes sure that upon touching the ground, the thruster is off and is not switched on again. Equations 22-24 would only be necessary for parameter values that would make the spacecraft bounce off after touchdown.

The dynamic behavior presented in figures 4-8 is generated by simulating the model including the controller with the proposed heuristic for 120 seconds (equations 1-17 and equations 18 and 24). The dynamic behavior of *Height* is given in Figure 4. Initially, the change in *Height* (i.e. *Velocity*) is relatively fast and, as the spacecraft approaches to the surface, the change in *Height* slows down. This behavior is comparable to the landing behavior of Apollo 15 (see Appendix). Hence, one can conclude that the behavior obtained by the control heuristic is a reasonable one; by a fast initial decline, the heuristic tries to decrease the time to land; by a slow final approach, it keeps the impact force well below harmful values. At the instant of touchdown, the value of *Velocity* is -0.05 meters per second (-0.18 km/h) creating a maximum impact force of circa 10,090 Newton, approximately 1.14 times the weight of the spacecraft on the target celestial body (8,870 Newton). The weight corresponds to the model variable *Gravitational Force*, which is the force that the landing gear must bear when the spacecraft is standing still on the ground. For example, if the value of *Velocity* was -3 meters per second (-10.8 km/h) at the instant of touchdown, the maximum impact force of circa 25,648 Newton, which is approximately 2.89 times the weight of the spacecraft on the target celestial body, would be generated by the impact. This example is just to give an idea about the success of the proposed heuristic and the importance of the effect of the final velocity on the maximum impact force. The discussion on the strength design of the spacecraft is beyond the scope of this study.

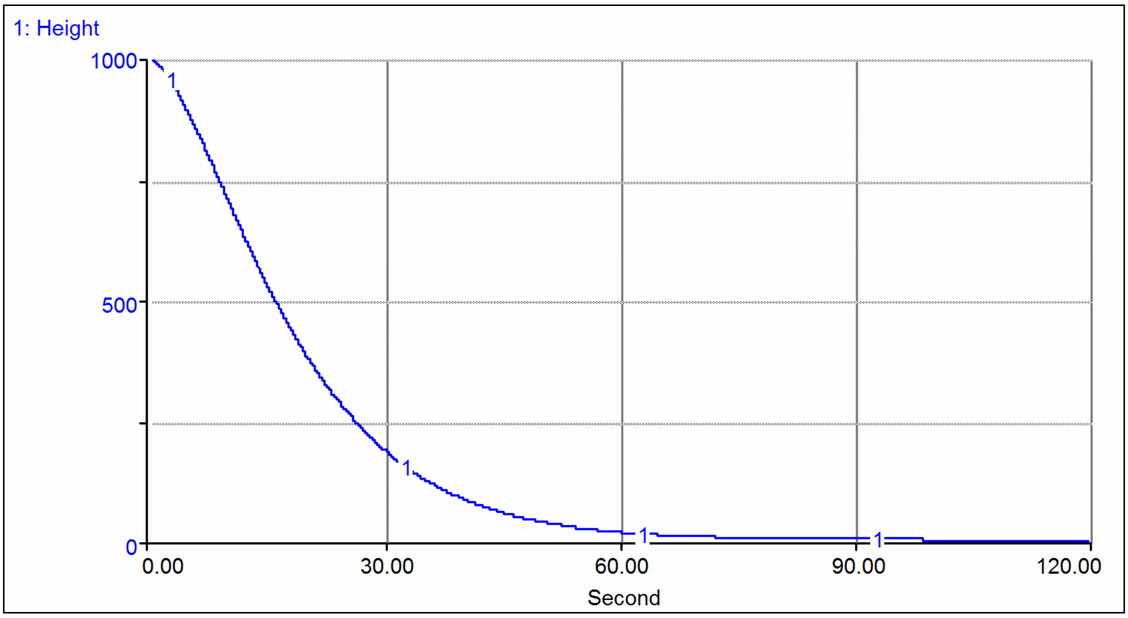

Figure 4: Dynamic behavior of *Height*

The dynamic behavior of *Velocity* and *Net Force* acting on the vehicle during landing are given in figures 5 and 6, which further explain the dynamic behavior obtained by the control heuristic. At first, the heuristic allows the spacecraft to accelerate in the negative direction towards the landing surface (see Figure 5, approximately within the time range of 0-10 seconds) by keeping *Net Force* negative (i.e. *Control Force* less than *Gravitational Force,* see figures 6 and 7). Aiming to decrease the duration of landing, *Velocity* continues to increase during this initial period. After this initial phase, *Velocity* decreases until the vehicle touches the surface (see Figure 5, approximately within the time range of 10-100 seconds). In this later phase, the heuristic produces more *Control Force* than *Gravitational Force* (Figure 7) resulting in a positive *Net Force* (Figure 6). At the moment of landing, *Control Force* is turned off (via the formulation of Desired Control Force given in Equation 22) and *Damping Force*, which is zero throughout the simulation up to this point, takes over and stops the vehicle (see figures 6 and 7, approximately around 100 seconds). The dynamic behavior of the variable named *Spring Compression* between seconds 98-102 is also plotted (Figure 8). *Spring Compression* shows a critically damped behavior as a result of the assumed landing gear design (see Equation 10 and Footnote 3).

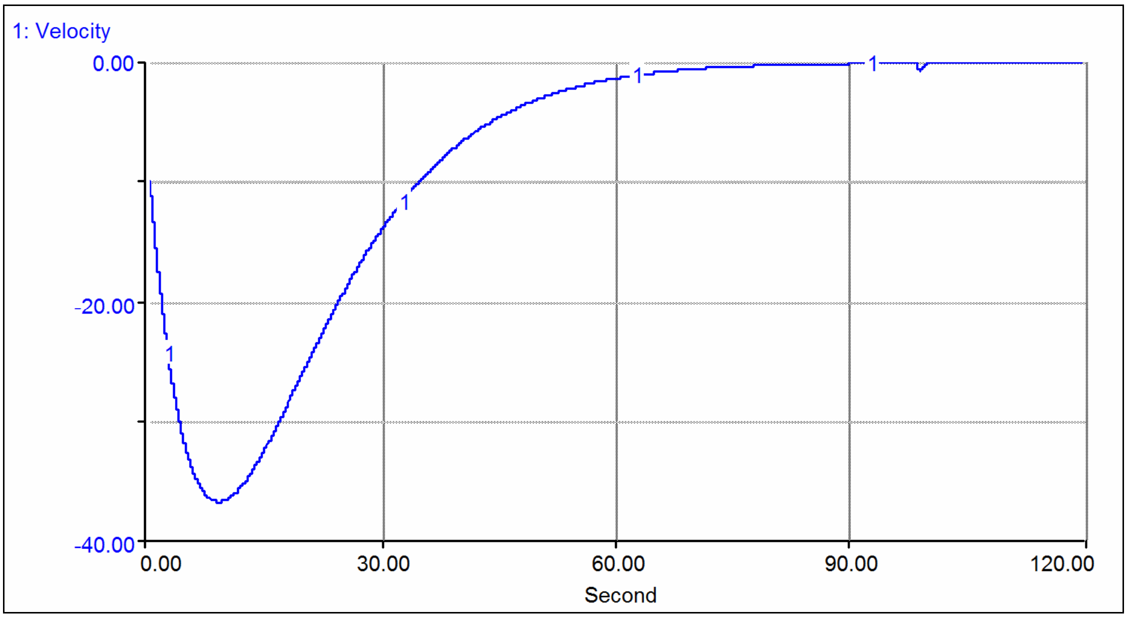

Figure 5: Dynamic behavior of *Velocity*

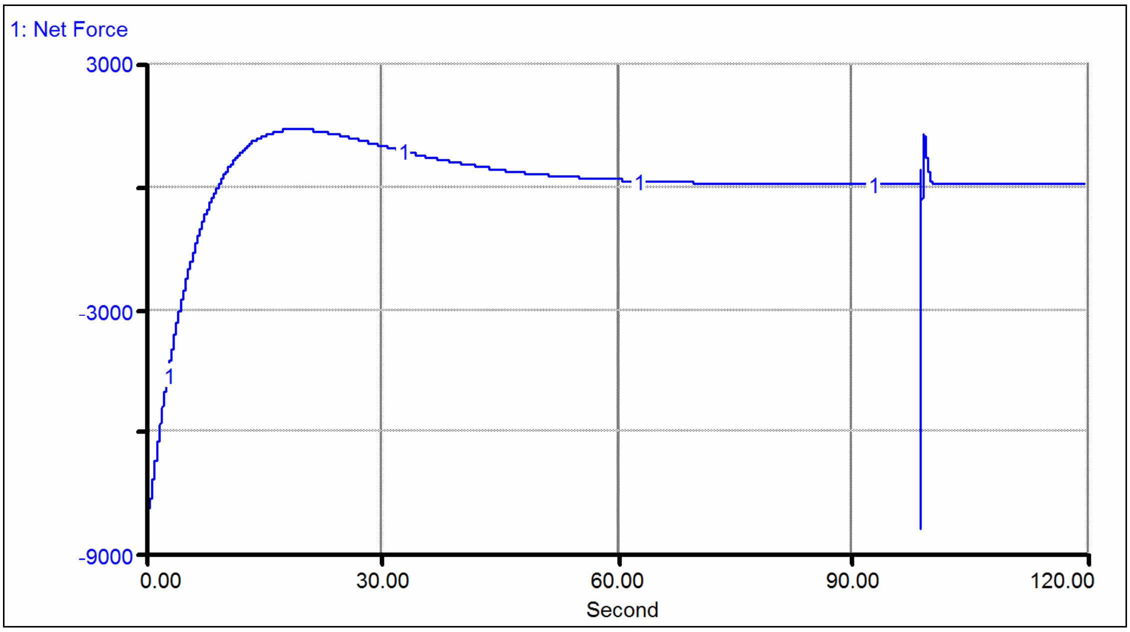

Figure 6: Net force acting on the vehicle during landing

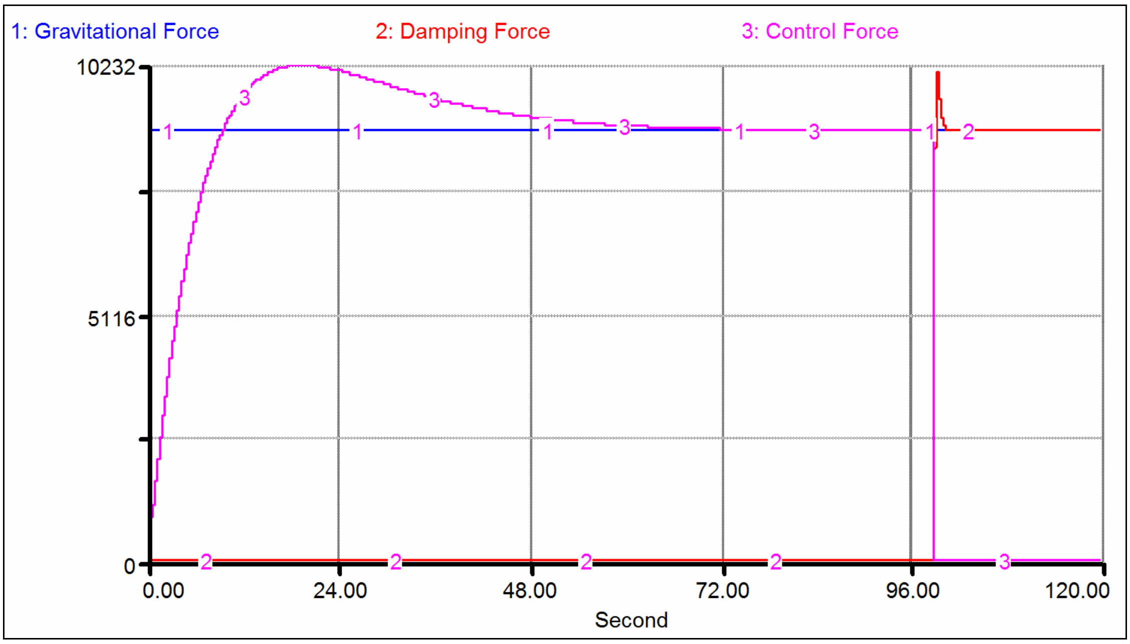

Figure 7: Absolute values of the forces acting on the vehicle during landing[5]

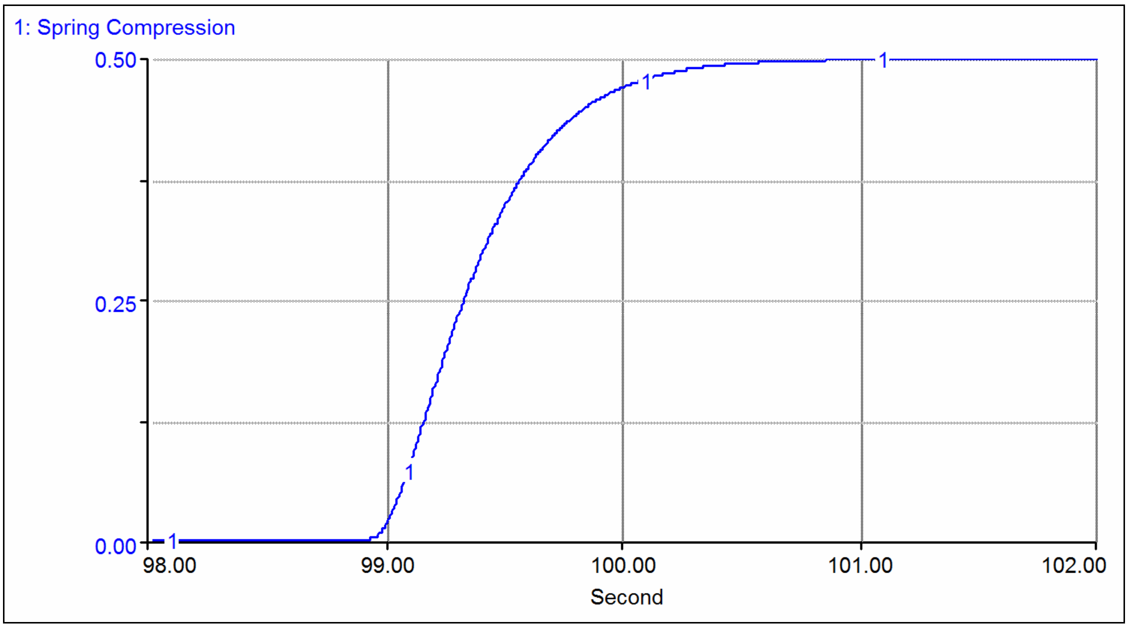

Figure 8: Dynamic behavior of *Spring Compression* during the final process of landing (between seconds 98-102)

[5] In order to ease the comparison of the different forces acting on the vehicle, the directions of the forces are ignored on this diagram.

4. Soft Landing Experimental Platform (SLEP)

We also developed an experimental simulation platform for soft landing (SLEP), based on the model and the heuristic. A screenshot of this simulation-based discovery learning environment can be seen in Figure 9. SLEP can be used to introduce simulation and dynamic complexity. It can also serve as an introductory control design tool as it allows to design the control heuristic and also the landing gear. The authors of this paper are willing to share SLEP given that it is going to be used for academic (non-commercial) purposes and given that proper credit will be given to the rightful owners.

The following parameter and initial conditions can be modified by the experimenter:

- *Mass*
- *Max Force*
- *Initial Height*
- *Initial Velocity*
- *Gravitational Acceleration*
- *Landing Gear Rest Compression*
- *Suspension Damping Factor*
- *Height Coefficient*
- *Control Force Damping Factor*

The following parameters are automatically calculated by SLEP after entering the above parameter and initial values and clicking on the “Run” button:

- *Suspension Spring Coefficient*
- *Suspension Damper Coefficient*
- *Velocity Coefficient*

Velocity Coefficient is calculated as given below:

$$\begin{pmatrix} Velocity \\ Coefficient \end{pmatrix} = \begin{pmatrix} Control\ Force \\ Damping\ Factor \end{pmatrix} \cdot \sqrt{\begin{pmatrix} Height \\ Coefficient \end{pmatrix} \cdot Mass} \quad \left[\frac{N \cdot s}{m} \right] \tag{25}$$

At the end of the simulation, the dynamic behaviors of *Height*, *Velocity*, and the forces constituting *Net Force* (i.e. *Gravitational Force*, *Damping Force*, and *Control Force*) are displayed as graphical outputs. SLEP also collects the following performance measures using the equations provided below:

$$Crash\,Time_0 = 0 \quad [s] \tag{26}$$

$$Crash\,Time_{t+DT} = Crash\,Time_t + \begin{cases} t/DT, & Crash\,Time_t = 0,\ Height_t < 0 \\ 0, & \text{otherwise} \end{cases} \quad [s] \tag{27}$$

$$Crash\ Velocity_0 = 0 \quad [m/s] \tag{28}$$

$$\begin{pmatrix} Crash \\ Velocity \end{pmatrix}_{t+DT} = \begin{pmatrix} Crash \\ Velocity \end{pmatrix}_t + \begin{cases} Velocity_t/DT, & Crash\,Velocity_t = 0,\ Height_t < 0 \\ 0, & \text{otherwise} \end{cases} [m/s] \tag{29}$$

$$Max\,Landing\ Force_0 = 0 \quad [N] \tag{30}$$

$$Max\,Landing\,Force_{t+DT} = \begin{pmatrix} Max \\ Landing \\ Force \end{pmatrix}_t + \begin{cases} \dfrac{\begin{pmatrix} Damping\ Force_t \\ -Max\ Landing\ Force_t \end{pmatrix}}{DT}, & \begin{pmatrix} Damping \\ Force \end{pmatrix}_t > \begin{pmatrix} Max\ Landing \\ Force \end{pmatrix}_t \\ 0, & \text{otherwise} \end{cases} [N] \tag{31}$$

$$Force\ Ratio = \frac{Maximum\ Landing\ Force}{Gravitational\ Force} \quad [\text{dimensionl ess}] \tag{32}$$

$$Max\ Acceleration_0 = 0 \quad [m/s^2] \tag{33}$$

$$Max\ Acceleration_{t+DT} = Max\ Acceleration_t + \begin{cases} \dfrac{\begin{pmatrix} Acceleration_t \\ -Max\ Acceleration_t \end{pmatrix}}{DT}, & Acceleration_t > \begin{pmatrix} Max \\ Acceleration \end{pmatrix}_t \\ 0, & \text{otherwise} \end{cases} [m/s^2] \tag{34}$$

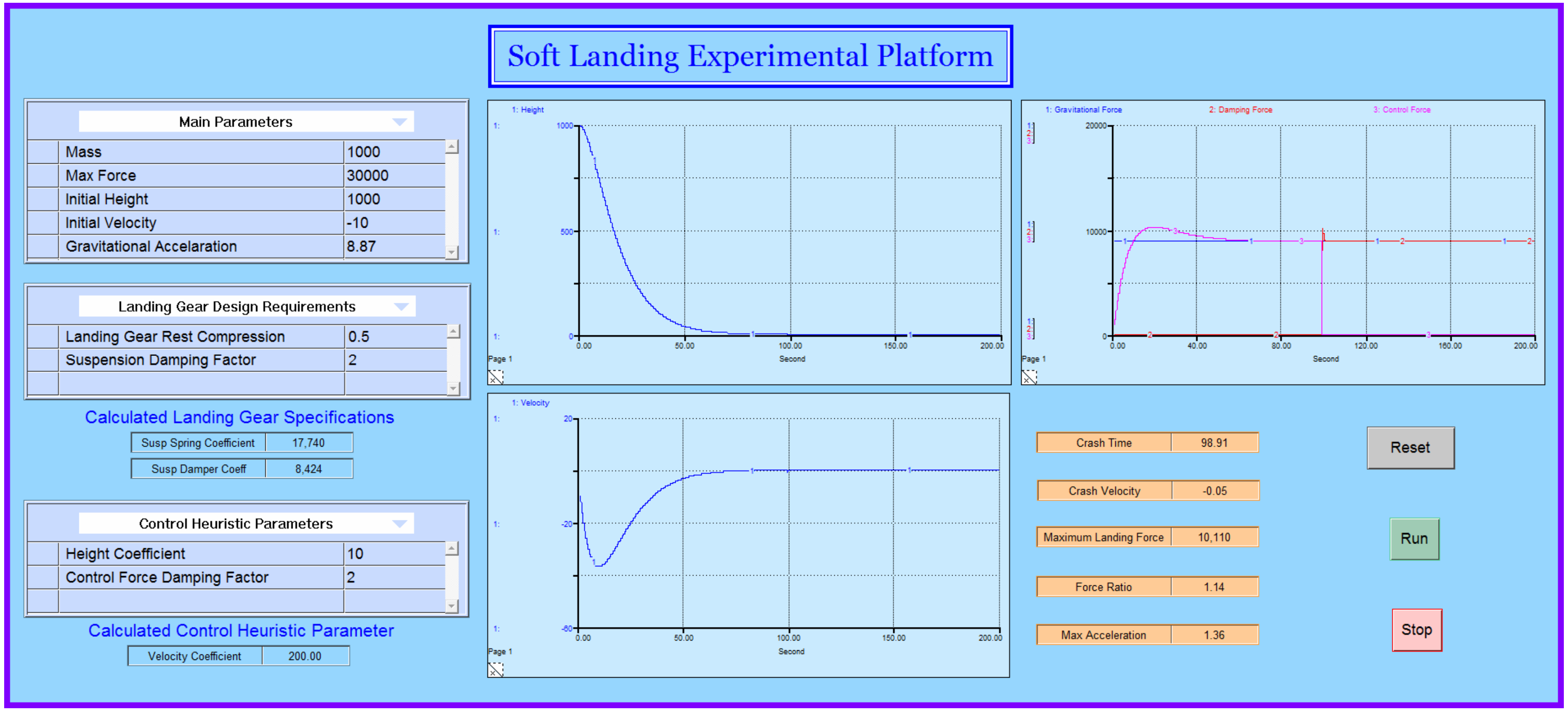

Figure 9: Screenshot of Soft Landing Experimental Platform (SLEP)

5. Conclusions and Future Research

In this study, we first developed a soft landing model using System Dynamics methodology. The modeling effort was focused on obtaining a valid and transparent representation of the soft landing challenge, which is to land the spacecraft as gently and as fast as possible. Besides model transparency, the model was kept as simple as possible in order to facilitate sharing and understanding of the model structure and dynamics, as one of the aims is to use the model in teaching. Even under the simplifying model assumptions, the soft landing problem still remains a challenging one. The main reason for the difficulty is that the control task requires simultaneous control of the height and velocity of the spacecraft, which have inertia and can only be indirectly affected by the reverse force thruster. The dynamics generated by the model is also explained in full detail.

We also developed a simulation-based discovery learning environment based on the model and the heuristic. The simulation model and Soft Landing Experimental Platform (SLEP) presented in this paper can be used to introduce simulation, modeling, and dynamic complexity to and as an introductory control design tool for physics, engineering, and interested social sciences students. It is also possible to develop a soft landing game based on the model as a platform for dynamic decision making experimentation.

Acknowledgement

We presented an earlier draft of this work in the paper titled "*A Soft Landing Model and a Mass Spring Damper Based Control Heuristic*" at the 29th International Conference of the System Dynamics Society, 24-28 July 2011, Washington DC – USA. We improved that previous work and divided it into two parts. One part is this paper and the other part is the paper titled "*Control Heuristics for Soft Landing Problem*", which is also presented at this conference.

References

Åström KJ, Murray RM. 2008. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press.

Atay, F.M. (ed.), 2009. *Complex Time-Delay Systems: Theory and Applications*, Springer, Berlin.

Barlas Y. 2002. System Dynamics: Systemic Feedback Modeling for Policy Analysis. *Knowledge for Sustainable Development - An Insight into the Encyclopedia of Life Support System.* UNESCO/EOLSS Publishers: Paris, France; Oxford, UK.

Forrester JW. 1961. *Industrial Dynamics*. Pegasus Communications: Waltham, MA.

Forrester JW. 1971. *Principles of Systems*. Pegasus Communications: Waltham, MA.

Hübler, A.W. 2007. Understanding Complex Systems, *Complexity* **12**(5) 9–11.

Liu XL, Duan GR, Teo KL. 2008. Optimal soft landing control for moon lander, *Automatica* **44**(4) 1097–1103.

Michiels W, Niculescu SI. 2007 *Stability and stabilization of time-delay systems: An eigenvalue-based approach*, SIAM.

Saysel AK, Barlas Y. 2006. Model Simplification and Validation with Indirect Structure Validity Tests, *System Dynamics Review.* **22**(3) 241–261.

Sterman JD. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*, Irwin/McGraw-Hill; Boston, MA.

Tanyolaç T, Yasarcan H. 2011. A Soft Landing Model and a Mass Spring Damper Based Control Heuristic, *Proceedings of The 29th International System Dynamics Conference*, Washington DC – USA.

Tanyolaç T, Yasarcan H. 2012. Control Heuristics for Soft Landing Problem, *Proceedings of The 30th International System Dynamics Conference*, St. Gallen, Switzerland.

Yasarcan, Improving understanding, learning, and performances of novices in dynamic managerial simulation games, *Complexity* **15**(4) (2010) 31–42.

Yasarcan H. 2011. Stock Management in the Presence of Significant Measurement Delays. *System Dynamics Review*. **27**(1): 91–109.

Yasarcan H, Barlas Y. 2005. A Generalized Stock Control Formulation for Stock Management Problems Involving Composite Delays and Secondary Stocks. *System Dynamics Review.* **21**(1): 33–68.

Zhang Y, Park JB, Chong KT. 2009. Controller design for nonlinear systems with time delay using model algorithm control (MAC), *Simulation Modelling Practice and Theory.* **17**(10) 1723–1733.

Zhou J, Zhou D, Teo KL, Zhao G. 2009. Nonlinear Optimal Feedback Control for Lunar Module Soft Landing, in IEEE (ed). *IEEE International Conference on Automation and Logistics*. Shenyang, China: IEEE.

Appendix

In plotting the dynamics observed in Figure 10, we connected to the Apollo 15 entry of the Wikipedia website (http://en.wikipedia.org/wiki/Apollo_15; accessed on 16 September 2011) and time coded the landing video on the page (http://en.wikipedia.org/wiki/File:Apollo_15_landing_on_the_Moon.ogg; accessed on 16 September 2011). Note that Apollo 15 was the fourth to land on the Moon (30 July 1971).

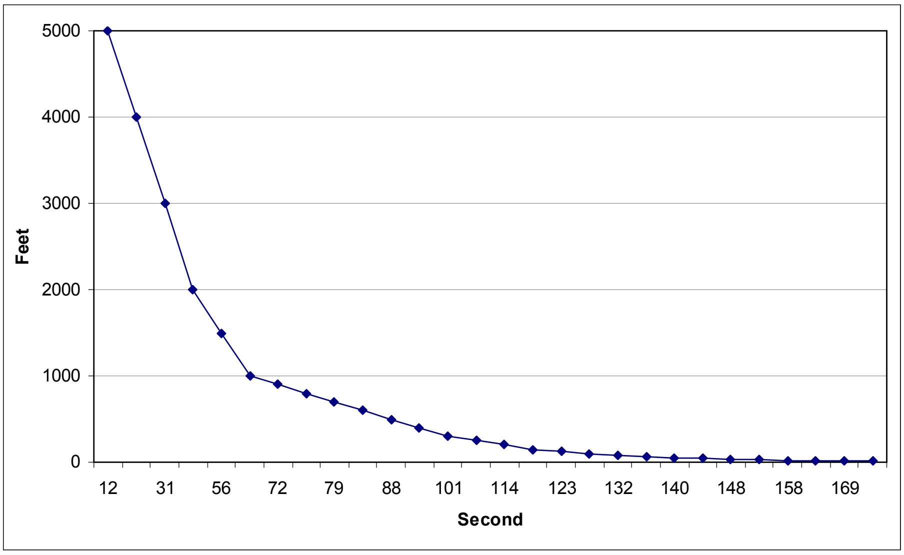

Figure 10: The landing dynamics of Apollo 15