Syracuse University

SURFACE

August 2019

A Feasibility Study of a Low-cost, Large-scale Thermal Treatment Process for Human Feces

Ryan John Homeyer
Syracuse University

Abstract

To hygienically manage the global sanitation crisis, it is pertinent to develop new methods for treating human excreta. This thesis proposes a new design where human feces is stored in a steel shipping container that is subject to shortwave solar radiation that, according to theory, heats the enclosed excreta to temperatures that inactivate fecal pathogens. The feasibility of this design is analyzed by way of numerical and experimental modelling. The experimentally validated model is used to simulate the effectiveness of the design over the course of many days of irradiation. This study shows that a sufficient temperature distribution (i.e., $T(x,y) > 46\degree C$) is not achieved within the enclosed feces after 200 days of irradiation but this cannot be concluded until a full-scale test of the prototype is completed. Models that utilize nighttime insulation predict that necessary temperatures are achieved after 117 to 136 days of irradiation.

A Feasibility Study of a Low-cost, Large-scale Thermal Treatment Process for Human Feces

by

Ryan J. Homeyer

B.S., Le Moyne College, 2017

Thesis
Submitted in partial fulfillment of the requirements for the degree of
Master of Science in Environmental Engineering.

Syracuse University
August 2019

Acknowledgments

Contents

List of Figures

List of Tables

Chapter 1: Introduction

As of 2015, 892 million people still practice open defecation worldwide (WHO, 2017). Further, 2.3 billion people lacked a basic sanitation service, defined by the World Health Organization (WHO) as a sanitation technology that minimizes human-excreta contact and is not shared with another household (WHO, 2017). As part of the United Nations Sustainable Development Goals (SDG), a continuation of the Millennium Development Goals conceived in 2000 "to reduce extreme poverty by 2015" ("Sustainable Development Goals", n.d.), the UN has stated that it seeks to: "By 2030, achieve access to adequate and equitable sanitation and hygiene for all and end open defecation, paying special attention to the needs of women and girls and those in vulnerable situations" (WHO, 2017). The 2017 WHO report, *Progress on Drinking Water, Sanitation, and Hygiene,* shows that the most need for improved sanitation facilities is in the lower latitudes, specifically in Africa as shown in Figure 1, while nations in higher latitudes tended to meet the endpoint of SDG 6 (WHO, 2017). Furthermore, current projections of national trends toward the elimination of open defecation show that this is not currently feasible by 2030, as shown in Figure 2. This conclusion is especially true for rural regions where fractions of the population practicing open defecation are significantly higher than their urban counterparts (WHO, 2017). This is suggestive of the necessity for different sanitation initiatives in rural, developing regions that can incentivize a transition toward basic sanitation technologies.
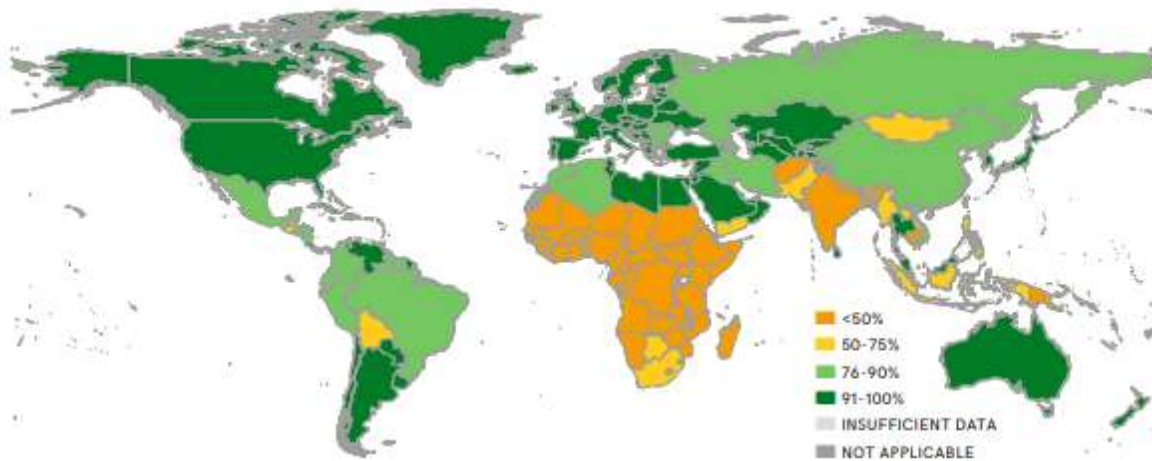
Figure 1: A map of global sanitation status by country from the 2017 WHO SDG Report (WHO, 2017).
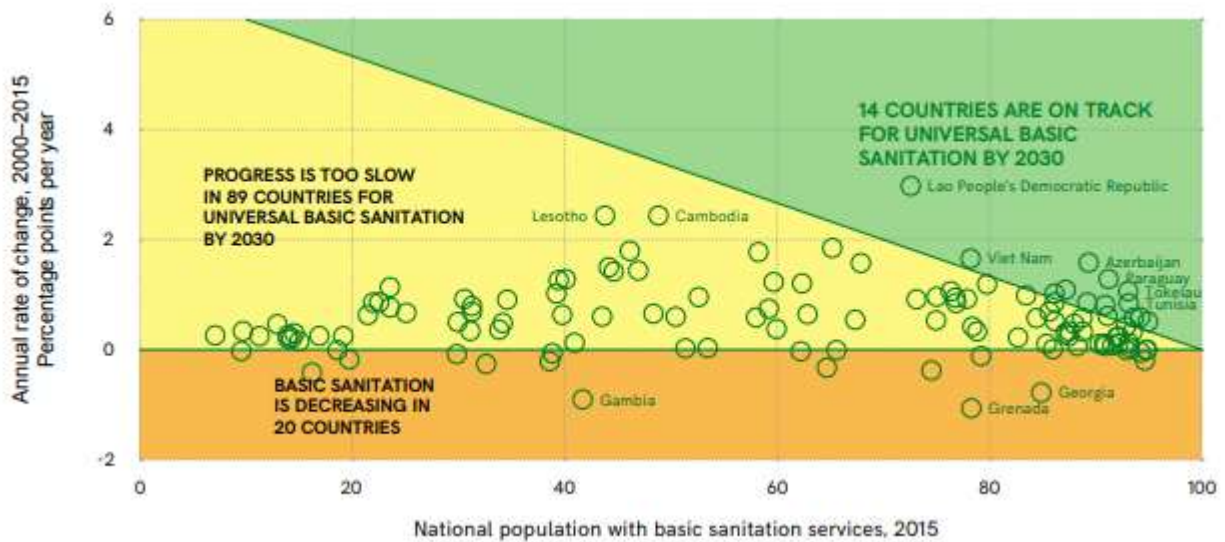
Figure 2: A plot from the WHO SDG report forecasting that the sanitation SDGs will not be met by 2030 (WHO, 2017).

Careful management of sanitation practices is crucial to ensuring public health in all communities because excreta, predominantly fecal matter, provide a vector for pathogenic infection (Feachem et al., 1983). Furthermore, poor sanitation practices contribute to high child

mortality in the developing world by causing diarrheal diseases that account for 15% of deaths for children under 5 years old (You et al., 2011). For example, diarrhea contributed to 41% of deaths of children under 5 in the Hartisheik A Camp in Ethiopia in 1989 (Davis & Lambert, 2002). These data show that in the case of both developing nations and refugee camps diarrheal diseases cause a significant portion of child deaths. Diarrheal diseases are most frequently considered as four organismal categories of concern: viruses, bacteria, protozoa, and helminths (Feachem et al., 1983). Viruses require a host in order to multiply and cause viral infections; however, they can survive for weeks without a host. Bacteria are the predominant cause of diarrheal diseases and, due to their widespread presence in human feces, bacterial species (e.g., *Escherichia coli*) are often used as an indicator of fecal contamination in water and soil. Protozoa can cause diarrheal diseases and dysentery. Helminths (parasitic worms) differ from other pathogens in that they do not multiply in the host. Rather, helminth eggs and larvae move among hosts via the fecal route and cause infections that are proportional to the number of helminthic worms that enter the host (Feachem et al., 1983).

Proper hygiene, including handwashing, limits one vector connecting the fecal-oral route of pathogenic transmission. But the construction and maintenance of hygienic sanitation facilities present another way to minimize sickness in a community. For example, open-air latrines (Mihelcic et al., 2009) and defecation fields (Davis & Lambert, 2002) allow for pathogens to reach humans by way of mosquitoes and other insects. Furthermore, open-air sanitation facilities can contaminate natural water bodies that are used for potable water and aquaculture.

Orner and Mihelcic (2018) provide an overview of several sanitation techniques and technologies that cover the spectrum from minimal isolation to centralized wastewater treatment. The typical categorical order of increasing isolation is: dig and cover, bucket latrine, ventilated

pit latrine, composting latrine (urine diverting and non-urine diverting), pour-flush toilet connected to a septic tank, and a toilet connected to a sewer system. The "dig and cover" method is a marginal improvement over open defecation because it ostensibly prevents direct contact by flies and humans after coverage but does not prevent direct human contact. Further, this approach does not sufficiently sequester the manure from erosive runoff generated by high intensity rainfall events. The bucket latrine system involves the collection of human feces in a bucket or container used to transport it to another end location (e.g., a pit or compost pile). This method still requires direct contact between the feces and the human being and is therefore considered unimproved by the authors (Orner & Mihelcic, 2018). The ventilated pit latrine, or its simpler cousin, the pit latrine, is a dug pit covered with a slab (and cap) that contains human feces and eliminates the need for contact between the human and the manure. The ventilated aspect of this technology refers to the addition of a pipe to promote the flow of malodourous air away from the hole where the user is located (Mihelcic et al., 2009). Composting latrines with urine diversion differ from those without diversion in that the solid excreta have a higher carbon-to-nitrogen ratio, which benefits the composting process and allows the urine to be used for agricultural purposes. The latrine without diversion of urine requires the addition of organic matter with a higher carbon-to-nitrogen ratio to yield a more productive compost operation. The pour-flush toilet connected to a septic tank collects excreta carried by a water input. This technology is useful because it collects the excreta in a rough tank, as opposed to a dug pit, that can be emptied and treated to the endpoint of being an innocuous effluent. Finally, the sewered toilet is a connection between excretion and an outflow pipe. The destination of the excreta is not always certain for sewered toilets as the effluent for some systems is transported to a wastewater treatment plant or process while others are immediately discharged into aquifers or surface water

bodies (Orner & Mihelcic, 2018). This overview shows that there exists a broad spectrum of sanitation techniques and technologies that can deal with excreta. Orner and Mihelcic (2018) classify them into three groups: unimproved technologies which do not yield a useful product, improved technologies that do not yield a useful product, and improved technologies that yield a useful product. "Dig and Cover" and bucket latrines are in the first group; ventilated pit latrines, pour-flush toilets with septic tanks, and sewered toilets are in the second group; and the composting latrines are in the third group (Orner & Mihelcic, 2018). Technologies that fall into the third group are especially ideal for sanitation initiatives in regions that are not currently slated to meet the 2030 SDG. They provide a way for a community to improve its public health and have an economically beneficial by-product of that effort.

Humans have used excreta effectively to provide nutrients to agricultural soil globally for centuries. However, in many places the excreta, deemed "night soil", have not been sufficiently treated prior to application (Jenkins, 2005). This can lead to sickness in both those doing agricultural labor as well as those selling and eating the produce grown with the untreated excreta (Feachem et al., 1983). The implementation of facilities in the third group in regions where the use of "night soil" occurs would serve to sanitize the fertilization infrastructure that is already in place.

Ecological sanitation, alternatively known as eco-sanitation or ecosan, is "an alternative low-tech approach to conventional wastewater systems" developed to address the impediments of intense water usage and significant startup cost of other sanitation solutions (Langergraber & Muellegger, 2005). These approaches, including composting latrines, involve the conversion of "human excrement into a pathogen-free soil amendment that can improve both the physical structure and the nutrient content of soils" (Mihelcic et al., 2009). Further, because the sanitation

problem is without a panacea, the UN recommends that eco-sanitation be heavily focused upon as a potential solution (Langergraber & Muellegger, 2005). The nutritional supplements provided to the soil by the treated excreta provide evidence that ecosan addresses both the sanitation problem and issues of food security and chemical pollution. The use of human excreta as a natural fertilizer rather than synthetic nitrogen serves to reduce contributions to the currently open human nutrient cycle (Langergraber & Muellegger, 2005). The human nutrient loop can be understood as the movement of nutrients from the natural environment into and out of human beings. The standard model is not a circuit, but rather a conduit through which nutrients pass from the soil into human beings (through agricultural products), whereupon through excretion enter natural water vis-à-vis wastewater treatment plants or direct dumping of wastewater (Jenkins, 2005). This presents a serious problem on two fronts because agricultural soils are often not replenished with the naturally balanced organic nutrients required for sustainable agricultural production, and natural water bodies are being polluted with excessive concentrations of fertilizers that promote many environmentally detrimental processes, including eutrophication (Guzha et al., 2005). The issue of anthropogenic nutrient flux out of the soil can be further thought of as a mass conservation problem with the soil as the control volume of interest. Nutrients that are naturally in the soil are taken up by agricultural products that are consumed by humans and ultimately flow into natural water resources. This effluent nutrient stream is balanced to achieve steady-state conditions by an influx of mined or synthesized fertilizers that are more costly than natural nutrient sources. However, this mass balance is unsustainable since the influx of artificial fertilizers cannot be maintained ad infinitum (Guzha et al., 2005). The adoption of ecosan on a large scale could effectively create steady state conditions for this mass flux because the nutrients that leave the soil in the form of harvested

produce would be able to reenter in the form of composted manure. Therefore, the use of ecological sanitation is beneficial on multiple fronts including, but not limited to, proliferation of safer sanitation practices, increased food security, and less pollution of natural water bodies.

Among eco-san technologies are those that utilize high temperatures to inactivate the pathogens in feces to render it safe for use as a soil amendment. Sanivation, a company that is working on alternatives to traditional sanitation technologies, has used thermal treatment to yield a reusable feces byproduct (Foote et al., 2017). However, this approach has only been implemented on a small-scale with significant required maintenance (Martin, 2016). It is desirable to implement the thermal treatment process on a large-scale for use in refugee camps and communities that currently lack basic sanitation services. One proposed design to achieve this goal is to fill a lined shipping container with human feces and allow incident shortwave solar radiation to heat the walls of the container, which would thereby heat the enclosed feces via conduction. The goal of this design is to isolate and inactivate the pathogens in the stored feces to render it less of a public health risk to the community at large. A secondary benefit of this design is the potential for the excreta to be used as a soil amendment after inactivation has been completed.

The human element inherent to ecosan is frequently a source of impediments to its successful implantation. It has been observed that public reception to the reuse of excreta varies—it is usually positive but can decline when the point of use would be agricultural food production (Simha et al., 2018). Additionally, proper use and maintenance of these systems requires community participation which can often lead to misuse, which has been observed in the case of composting latrines (Naughton et al., 2018). It will be important to engineer the implementation of this design with care so that it can be used for the long term. This would

require public acceptance, education, and a rigorous, easy-to-follow maintenance scheme. Further, acceptance and uptake of ecosan has been correlated to an increase of lower-cost technologies that can be constructed with ubiquitous materials (Tumwebaze et al., 2011). This observation supports the potential for the success of the proposed design since shipping containers can be accessed globally and are low-cost relative to other treatment technologies.

The key engineering design consideration for systems that utilize high temperatures is that inactivation does not occur for systems that do not reach $45°C$ irrespective of the exposure time (~318K) (Ruiz-Espinoza et al., 2012). Alternatively, it has been observed that systems that maintain a temperature of $42°C$ (~315K) for a period of at least 6 months to 1 year are sufficient for total pathogen inactivation (Mihelcic et al., 2009). Additionally, WHO has claimed that all fecal pathogens will be inactivated after one week when maintained at a temperature of $46°C$ (~319K) (Jenkins, 2005). Since there is no singular, agreed-upon temperature requirement for total pathogen inactivation, these systems require a conservative design temperature and a factor of safety. This is especially true since the goal of sanitation is to reduce the viability of the fecal disease vector. Therefore, this analysis considers the lower bound for successful inactivation of pathogens will be the long-term achievement of temperatures no less than 319K throughout the entire excreta containment vessel.

Chapter 2: Methods

A thorough investigation of the proposed thermal treatment system was carried out for this thesis using an experimentally validated, physics-based numerical model. The analyzed design consists of a cargo shipping container coated with a low albedo paint, a liner like those used in the transport of bulk dry goods, and a volume of human feces equivalent to that of the

liner which is dependent upon the container size. Such a liner and container system are shown in Figure 3.

Figure 3: The proposed thermal treatment system that is analyzed in this thesis utilizes a lined shipping container holding manure until fecal pathogens are sufficiently inactivated by solar heating(image source: http://www.eceplast.com/liner-for-container/fluid-liner/).

A key design criterion for the logistical development of this system is the time for a container to be filled with feces. The supply chain considerations with respect to this design require that the filling of the container and the storage of the feces do not impede the basic sanitation services that are being utilized by the community of concern. Towards investigating this concern, it is important to consider the rate at which a container would be filled relative to its required treatment time. The former can be calculated using Equation 1, where dV_F/dt is a volumetric feces generation rate.

$$\text{FillTime} = \frac{\text{Vessel Volume}}{\left(\frac{dV_F}{dt} \cdot \text{population}\right)} \tag{1}$$

Assuming a negligible difference between the vessel (i.e., liner) volume and the cargo container volume, the vessel volume can be computed as the product of the dimensions of the container.

The numerical model is used to approximate the thermodynamics for an ISO Sea Box container of dimensions 2.6 m x 2.6 m x 6.0 m, the volume of the container is 40.56m^3, or 40,560L. Utilizing an average feces generation rate of $dV_F/dt = 0.24$ L d^{-1} per capita (Murphy, 2015), the time required to fill the vessel by a 1000-person community was calculated to be 169 days. Since this is on the order of desired treatment time for a shipping container of this size, it would be necessary to supply at least two shipping containers to each group of 1000 individuals. Since feces generation rates, population, and container size can vary greatly, it would be important to develop empirical loading rates for each implementation site.

Other important design parameters in this concept include energy, water, and maintenance requirements for operation. Since the treatment only requires ambient exposure to solar radiation, there are no external energy requirements. Treatment of the manure does not require water, but it would be necessary to install a hygiene station with access to potable water at the implementation site. Use of this station should be required of those that come into direct contact with the manure to maintain pathogenic isolation. Filling and installing the container at the implementation site would require an initial input of labor. This would likely be significant at first, especially regarding movement of the container as well as loading it with feces. Filling and installation must also account for the placement of the vessel in a location where it minimizes danger to the surrounding community as well limits negative impacts on quality of life in the surrounding community. Some maintenance would be necessary to check for failures in the liner as well as leakages from the container but, with these exceptions, the system is low maintenance.

2.1: The Conceptual Model

The conceptual model for the heating of the proposed design is shown in Figure 4. Development of the conceptual and numerical models utilized Schroeder's *An Introduction to*

Thermal Physics, Hagen's *Heat Transfer with Applications*, and Sauer's *Numerical Analysis*

heavily.
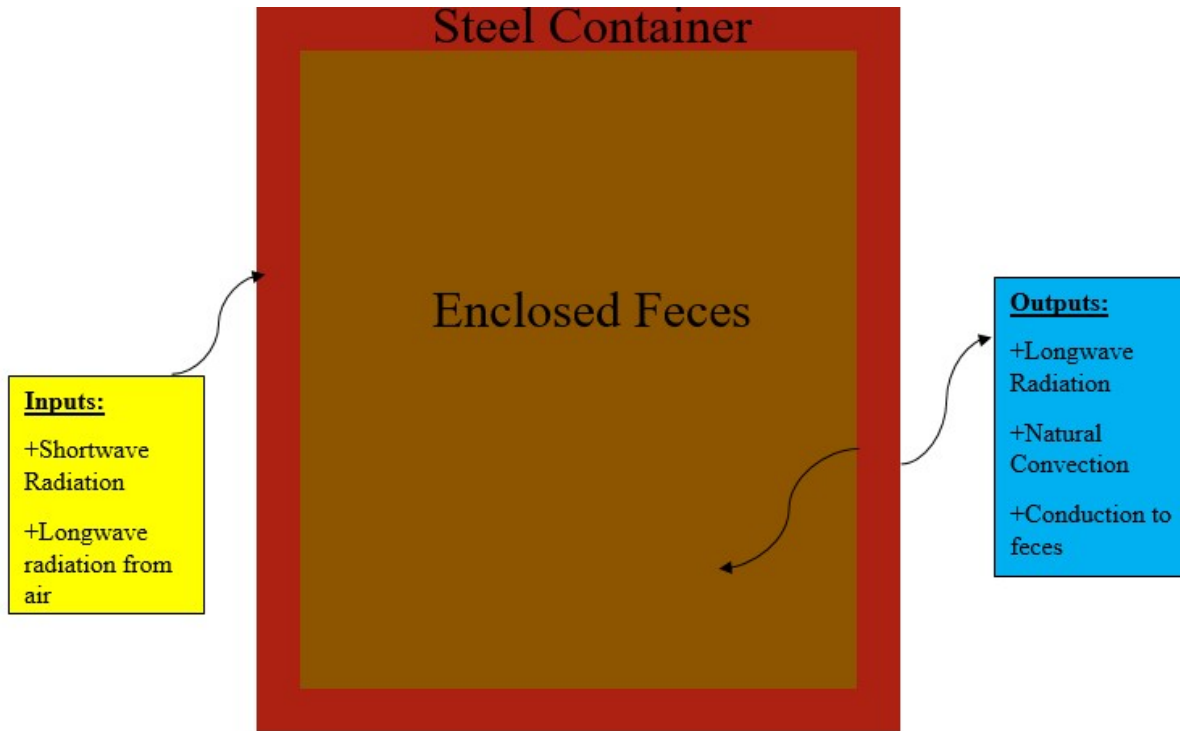
Figure 4: The conceptual model of heating for the prototypal design incorporates heating of the steel by shortwave solar radiation and longwave radiation from the air. Heat is lost by longwave radiation, natural convection to the air, and conduction to the enclosed feces. The enclosed feces is heated via conductive heat transfer from the steel.

Energy is delivered to the wall of the container as incident shortwave solar radiation. The amount

of heat absorbed by the steel wall is a function of the albedo of the steel. A fraction of heat is

returned to the atmosphere as longwave radiation and can be determined by the Stefan-

Boltzmann Law. The remaining heat is transferred to the enclosed feces via conduction, given by

Fourier's Law. Finally, heat is lost by convection to the atmosphere. The hot wall of the

container sets up a buoyancy-driven thermal boundary layer in the adjacent fluid resulting in

convective heat loss from the steel to the air (Ostace et al., 2013). The heat flux per unit area that

provides a conceptual model for this system is given by Equation 2:

$$\dot{q} = S(1 - \alpha) - \sigma\varepsilon\left(T_s^4 - T_A^4\right) - h_{Air}^{NC}(T_s - T_A) - k_F\frac{(T_S - T_F)}{\delta x} \qquad (2)$$

After the sun has set, the insolation to the shipping container reduces to zero, which yields the

nighttime heat transfer relation shown in Equation 3:

$$\dot{q} = -\sigma\varepsilon\left(T_S{}^4 - T_A{}^4\right) - h_{Air}^{NC}(T_S - T_A) - k_F\frac{(T_S - T_F)}{\delta x} \qquad (3)$$

Where:

$$\dot{q} \equiv \text{Heat Flux per unit area} \qquad (4a)$$

$$S \equiv \text{Incident shortwave solar radiation} = 1000\,\frac{W}{m^2} \qquad (4b)$$

$$\alpha \equiv \text{Albedo} = 0.13 \qquad (4c)$$

$$\sigma \equiv \text{Stefan} - \text{Boltzmann Constant} = 5.67 \times 10^{-8}\,\frac{W}{m^2K^4} \qquad (4d)$$

$$\varepsilon \equiv \text{Emissivity} = 0.87 \qquad (4e)$$

$$T_S \equiv \text{Steel Temperature} \qquad (4f)$$

$$T_A \equiv \text{Air Temperature} \qquad (4g)$$

$$h_A^{NC} \equiv \text{Natural convection coefficient for air} \qquad (4h)$$

$$k_F \equiv \text{Thermal conductance for feces} = 0.5\,\frac{W}{m \cdot K} \qquad (4i)$$

$$T_F \equiv \text{Feces Temperature} \qquad (4j)$$

$$\delta x \equiv \text{Characterisitc conduction length} \qquad (4k)$$

Equations 2 and 3 show that the steel wall of the container primarily gains heat by a constant

shortwave radiation intensity, S, and primarily loses heat as a quartic power of its temperature

(i.e., Stefan-Boltzmann Law). Heat is also lost from the steel wall by convection and conduction

as a function of the temperature difference between the steel and the air (convection) and the

steel and the enclosed feces (conduction). Convective and conductive losses are related to the

temperature difference between the two media as well as a heat transfer coefficient. This coefficient is the thermal conductivity for conductive heat transfer and is a tabulated property of the conductive medium (Hagen, 1999; Pollution Research Group, 2014). The convective heat transfer coefficient is calculated using empirical relationships provided in Hagen (1999). There is also a component of heat transferred to the steel as a result of radiation from the air that is calculated as a quartic power of the air temperature.

The heat transfer relation and its components provide a means of predicting the temperature of the steel container over the period of service, thus yielding the temperature of the steel as a function of time (i.e., $T_s(t)$). Utilizing this result as a boundary condition, the spatiotemporal temperature distribution can be predicted using the heat equation given in Equation 5:

$$\frac{\partial T_F}{\partial t} = \alpha_F \nabla^2 T_F \qquad (5)$$

Where:

$$\alpha_F \equiv \text{Thermal diffusivity of feces} = 1.71 \times 10^{-7} \frac{m^2}{s} \qquad (6)$$

This differential equation is useful in this analysis because it relates the evolution of the temperature in time (LHS) to the evolution of the temperature in space (RHS).

These equations account for the predominant heat transfer mechanisms in this system in order to maximize simplicity and present a system that can be reproduced in a laboratory environment. This conceptual model does not incorporate forced convection (i.e., wind), relative humidity, or the diurnal variability of air temperature. These factors are not considered because of their extreme temporal and geographic variability. Further, each of these variables is difficult to adjust in the laboratory setting that was used to validate the numerical model with a small-

scale experiment. Additionally, the conceptual model simplifies the geometry of the shipping container by assuming that the doors and corrugated walls are planar, perfectly connected, steel plates without any structural deformities. The container doors and the heat flux through their imperfections were not considered in this analysis.

The storage of the manure in a closed shipping container will likely encourage anaerobic digestion and the production of biogas. The effect of biogases on the heat transfer model are not considered. Additionally, the internal generation of heat by microbial activity is considered negligible in this conceptual model, which is a typical assumption in the modeling of anaerobic digesters (Rennie et al., 2017).

The conceptual model was further simplified by dividing the assumed total shortwave heating time of twelve hours into three periods of four hours of constant, directly incident shortwave heating per side. The incidence of the shortwave radiation was considered to be normal to the wall of the container during each heating period. This is a rough approximation of the passage of a daily irradiance cycle of the shipping container in the field. This approximation allowed straightforward, programmable calculations and replicable laboratory experiments to be conducted. Following the period of insolation, the conceptual model is best described by Equation 3, which describes heat transfer without incident shortwave solar radiation. Additionally, the value of $1000W/m^2$ is used to represent the shortwave solar radiation at the ground surface. This value will vary depending upon the time of year and location of use but was simplified to a constant value for the validation experiments. Additionally, the sides of the container will almost certainly receive less normally incident radiation than the top due to the varying solar azimuthal angle. Estimating the extent to which the radiation will vary among the different faces is difficult so only a constant irradiation intensity was considered.

2.2: The Numerical Model

A physics-based numerical model of the proposed design was developed in MATLAB[1] based on the set of equations developed above. A two-pronged approach was taken in order to predict the evolution of this system under the physical conditions set forth by the conceptual model. The first leg of the solution method was completed with a cyclic scheme where the heat transfer relation was integrated with respect to the steel temperature to yield the time required, δt, to increase T_S by a small amount, δT. The integrals were computed using MATLAB's *integral* command. This scheme is presented in Equations 7 and 8 (Nave, 2000).

$$\dot{q}A = mC\frac{dT}{dt} \tag{7}$$

$$\delta t = \int dt = \frac{mC}{A} \int_{T_S}^{T_S+\delta T} \frac{dT}{\dot{q}} \tag{8}$$

Where[2]:

$$m \equiv \text{Mass of the steel wall} = \rho_S V_S \approx 240\text{kg} \tag{9a}$$

$$C \equiv \text{Specific heat of steel} = 470 \ \frac{J}{kg \cdot K} \tag{9b}$$

$$A \equiv \text{Area of the steel wall} = 15.6\text{m}^2 \tag{9c}$$

The steel temperature at the end of the time interval is used as a boundary condition[3] along with the time interval, δt, to perform a one-dimensional finite difference approximation of the heat equation to yield the new temperature of the feces at a distance, δx, from the wall. The

---

[1] For practicality, an analytical solution to this problem was not sought.
[2] Material properties were averaged using values provided in Hagen's *Heat Transfer with Applications*.

[3] The other boundary condition is the temperature of the far side steel wall, which is assumed to be the initial temperature of the steel.

discretization of the heat equation for one spatial dimension, Equation 10, into a one-dimensional

finite difference scheme is shown in Equation 11.

$$\frac{\partial T}{\partial t} = \alpha_F \frac{\partial^2 T}{\partial x^2} \tag{10}$$

$$T_i^{t+\Delta t} = T_i^t + Z(T_{i+1}^t + T_{i-1}^t - 2T_i^t) \tag{11}$$

Where:

$$Z = \frac{\alpha_F \Delta t}{\Delta x^2} < \frac{1}{2} \tag{12}$$

The figure below shows the discretization scheme used to determine the relation given in

Equation 11:
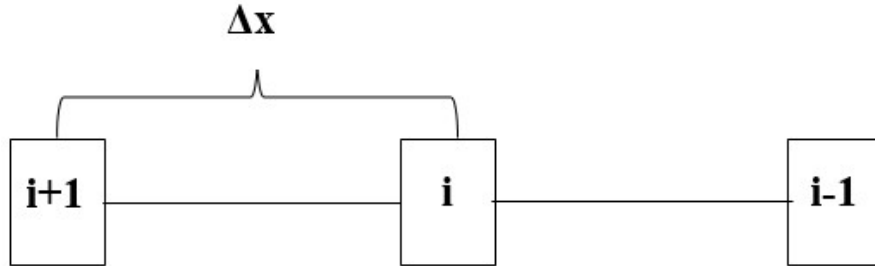
Figure 5: The one-dimensional discretization scheme utilized for approximating the rise of T_F in conjunction with the rise of T_S.

As shown in Figure 5, i represents the nodal locations separated by a distance, Δx. Both Δx and

Δt are subdivisions of the spatial and temporal ranges utilized in this scheme. Δx is a single

subdivision of the width of the box[4]. Δt is a single subdivision of the time interval, δt. When

substituted into Equation 12, these step sizes must yield $Z < 0.5$, which is a result of a Von

---

[4] In this model, and in most shipping containers, the height and width are equivalent.

Neumann stability analysis for a one-dimensional finite difference approximation (Anderson, 1995).

The distance δx is described in Equation 4k as the characteristic length of conduction and was set as twice the spatial discretization step within the fecal continuum. The rationale for this choice was based on physical intuition that the layer of feces closest to the steel container wall would provide the most conductive feedback to the wall. Once a new T_F and T_S were determined, then the heat transfer relation and the natural convection coefficient were updated before recycling through this scheme.
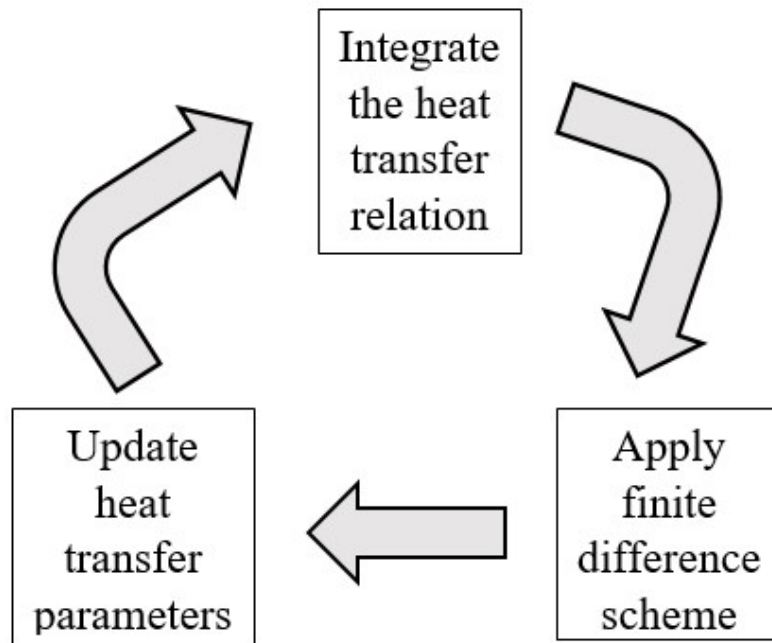
Figure 6: The solution scheme for the first leg of the numerical model. Integration of the heat transfer relation as shown in Equation 8 is followed by the 1-D finite difference approximation shown in Equation 11 to determine $T_S(t+\delta t)$ and $T_F(\delta x,t+\delta t)$. Following these calculations, the natural convective heat transfer coefficient and the heat transfer relation are updated.

The iterative integration scheme was utilized to determine the temporal evolution of the steel temperature so that the heating of the feces could be predicted. Three periodic boundary conditions (BCs) were determined and used in a two-dimensional numerical approximation of the heat equation. The three BCs are the temporally varying temperatures of the right side wall,

the top wall, and the left side wall that each experience sunlight from $t \in [0,14400]$s, $t \in$ [14400,28800]s, and $t \in [28800,43200]$s, respectively. The only other difference between these three surfaces is that the calculation of the natural convection coefficient included in the heat transfer relation differs between the vertical (i.e., side) walls and the horizontal (i.e., top) wall. Because natural convective heat transfer from the steel to the atmosphere is predicted to be less than other portions of the heat transfer relation, this should not yield a very significant difference between the three boundary conditions. Other than the time lag, the temperatures of the two side walls are predicted to be identical.

The two-dimensional solution of the heat equation was carried out to yield a spatiotemporal temperature distribution (i.e., $T(x, y, t)$) within the continuum of feces. Since the model posits that sunlight is directly and equivalently incident on all parts of the container, the temperature distribution is not predicted to vary significantly along the longitudinal axis of the shipping container (i.e., $\frac{\partial T}{\partial z} \approx 0$). This assumption is not fully correct since there is the potential that the end caps would act as heat sinks, especially on the end with the doors. These fringe effects were not studied numerically or experimentally, but their negative impact on the heating of the feces could be rectified by insulating the ends of the container. Therefore, for the second leg of the numerical study, the heat equation simplifies to the form shown in Equation 13, with a discretization relation in Equation 14.

$$\frac{\partial T}{\partial t} = \alpha_F \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \tag{13}$$

$$T_{i,j}^{t+\Delta t} = T_{i,j}^t + \beta \left(T_{i+1,j}^t + T_{i-1,j}^t + T_{i,j+1}^t + T_{i,j-1}^t - 4T_{i,j}^t \right) \tag{14}$$

Where:

$$\beta = \frac{\alpha_F \Delta t}{\Delta x^2} < \frac{1}{4} \tag{15}$$

The figure below illustrates the discretization scheme used to reach Equation 14:
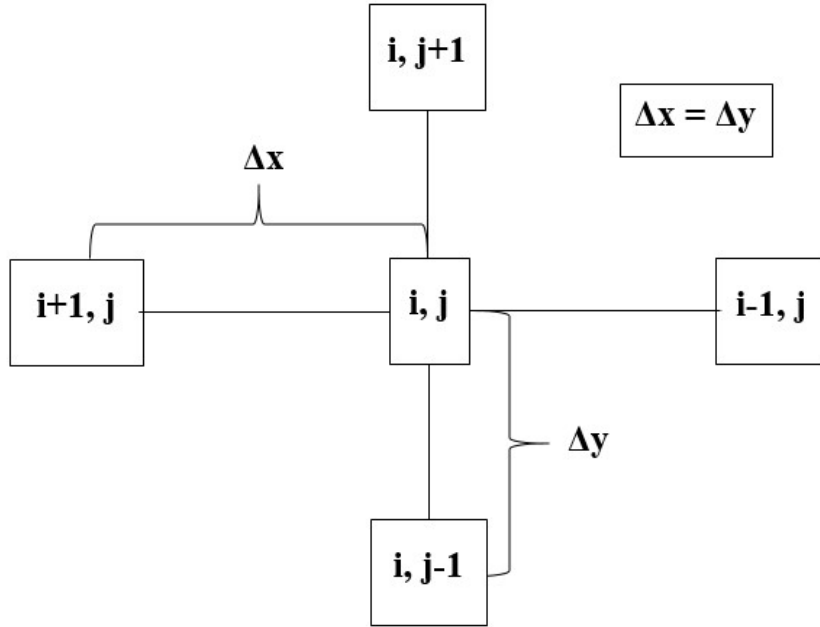
Figure 7: The two-dimensional discretization scheme utilized for approximating the rise of T_F throughout the entire container over a simulation of many days.

Figure 7 shows the nodal locations, given by i and j, within the fecal continuum. Again, Δx and Δt are subdivisions of the spatial and temporal intervals for this analysis. Further, the limiting condition of the increment intervals is given in Equation 15 and is the result of a Von Neumann stability analysis (Anderson, 1995) for a two-dimensional finite difference approximation. Furthermore, a single spatial increment, Δx, is used in this solution scheme because $\Delta x = \Delta y$ due to the symmetry of the domain.

Utilizing this solution scheme in conjunction with the periodic boundary conditions calculated during the first leg of the solution allows for a long-term simulation of the proposed design. This, combined with experimental validation, allows for a robust numerical model that

can predict the effectiveness of this design with respect to the time temperature requirements for the inactivation of fecal pathogens.

2.3: Scale Model for Model Validation

In order to verify the numerical model of the proposed design (i.e., feces in direct conductance with the walls of the container), a scale model that is geometrically similar to the conceptual model was developed. The scale model of the shipping container is a 0.076 scale replica of the lined shipping container. The small-scale design will be referred to as the model and the full-scale design will be referred to as the prototype for the rest of this thesis. The 0.1976m x 0.1976m x 0.456m model was formed from 24-gauge galvanized steel and then coated with "Red Oxide" spray primer (Rustoleum Inc, Vernon Hills, IL) to decrease the surface albedo.
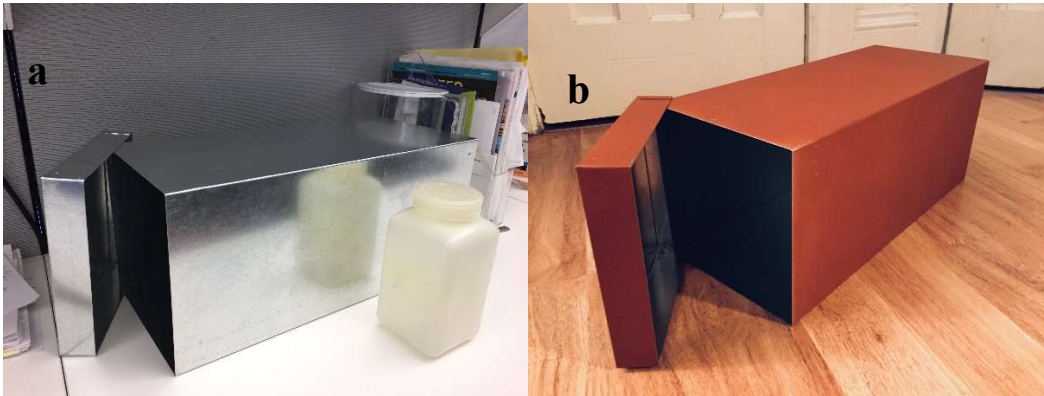
Figure 8: The scale model before(a) and after(b) painting. The end cap of the model is 0.1976m x 0.1976m. The side walls are 0.1976m x 0.456m.

In order to scale down the proposed design, it was necessary to simulate the container liner and the enclosed feces. The container liner was simulated using a polyethylene-based trash bag. The enclosed feces was simulated using a mixture developed by the Pollution Research Group at the University of KwaZulu-Natal (UKZN) in Durban, South Africa (Pollution Research Group, 2014). The recipe, provided below, was developed as part of the Reinvent the Toilet Fair

sponsored by the Gates Foundation. This fecal simulant was desirable for the lab-scale experiment based on the similarity in thermal conductivity and viscosity to feces.

Table 1: Simulant Recipe	
Ingredient	% Required
Yeast	7.3
Cellulose	1.2
Polyethylene Glycol	2.7
Psyllium	2.4
Peanut Oil	3.9
Miso	2.4
$Ca_3(PO_4)_2$	2.4
Water	77.6

Figure 9: Approximately 2kg of "cooked" fecal simulant using the UKZN recipe.

An important note on the production of the fecal simulant: since there is a large amount of yeast used in the recipe, it is prudent to allow the simulant to rise before putting it in a container as there is the potential for overflow and/or the buildup of undesired pressure. The scale model was designed so that simulant could be added to the model through a removable end cap.

In following the conceptual model proposed in Figure 4 and Equations 2, 3, and 5, it was important that the scale model of the prototype maintain geometric similarity. Geometric similarity is defined in a way that the dimensions of the model are related to the prototype by a constant scalar multiple. In this case, the scalar multiple that relates the length of the prototype to

the length of the scale model is 0.076. This singular scale factor allows for the validation of model outputs utilizing experimental measurements because the physics of conductive heat transfer relies upon the separation of hot objects from less hot objects (as seen in Equation 5). If geometric proportionality is maintained and conduction is the driving heat transfer mechanism, then an experimentally validated numerical model should be able to be used to make predictions about prototypes of a range of sizes (Skoglund, 1967; Tan, 2011). This scale factor is not applied to the duration of heating because the temporal similarity in conduction problems goes as the inverse square of the scale factor (Skoglund, 1967). This scaling would yield very short heating periods[5] that would not be suitable for making multiple, consistent temperature measurements. Instead, similarity between measured temperatures and predicted temperatures is utilized as validation of the model.

The two measurements that are predicted in the numerical simulation are $T_S(t)$ and $T_F(x, y, t)$; therefore, it was necessary to measure these during the model validation experiments. The steel temperature was measured using an infrared thermometer[6]. Over the temperature range of interest, the uncertainty was $\mp 2\,°C$. The temperature within the fecal simulant was measured using waterproof digital temperature sensors (DS18b20, Elenker, California) at a one-minute interval. The sensors were connected to an Arduino UNO (Uno Rev 3, Arduino, Italy) and then to a laptop computer, and these data were recorded, with timestamps, to a .txt file using the program CoolTermWin. Sensors were uniformly inserted into the simulant to a depth of 20 cm at a spacing of 5 cm horizontally and 4 cm vertically to form a rectilinear measurement grid.

---

[5] The heating duration would be on the order of 80 seconds.
[6] A General IRT207 was used to measure the steel temperature.

Figure 10: The sensor grid before (a) and after (b) the addition of the sensors.

The sensor locations within the grid would then be used to relate the measurements that were made to those predicted by the two-dimensional finite difference approximation. The sensor cables exited the model through a 1.5-inch hole in the end cap. The open space in the end of the container was insulated with bubble wrap. The hole in the end cap was insulated using plumber's putty.

The experiment was driven by two high pressure sodium (HPS) greenhouse lamps[7] that generated a broad-spectrum light largely in the visible range. The lamps were attached to a support structure and rotated about the model during the model validation experiments at a distance that maintained intensity of the shortwave radiation near 1000 W/m^2. This distance (12.1cm) was determined from measurements with a 4-component net radiometer (Apogee Scientific SN-500, Logan, UT).

---

[7] Hydroplanet 150W HPS Plug and Play Grow Lamp
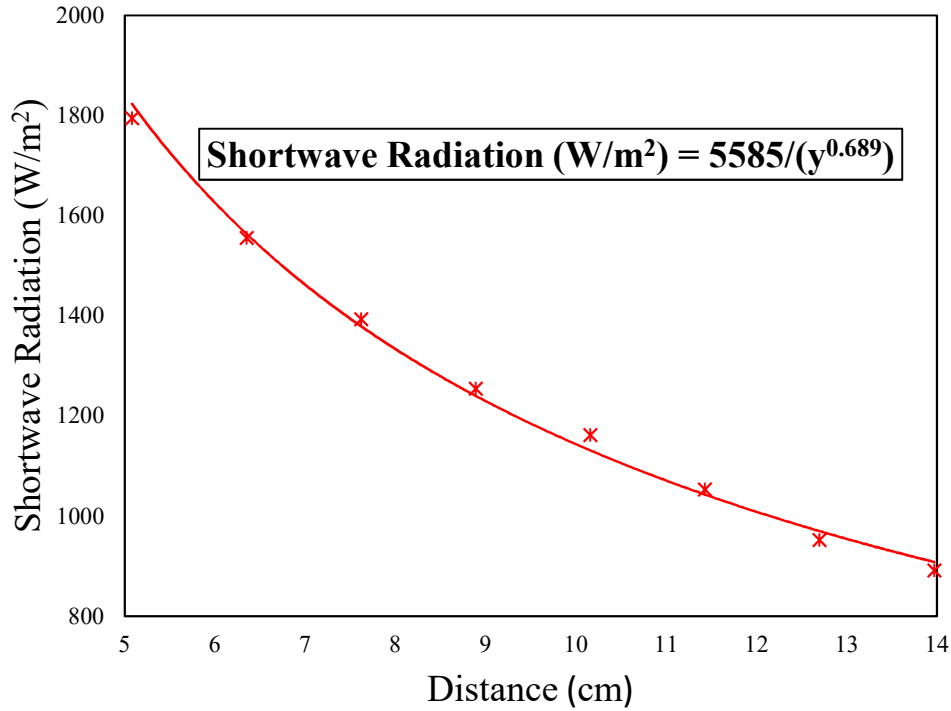
Figure 11: A plot of the measured shortwave radiation as a function of distance away from the lamp. Measurements were made with the Apogee Scientific SN-500 Net Radiometer at fixed distances away from the lamp.

During the experiment, the three exposed sides of the box were separately illuminated for 680 s each. Afterward, the light was turned off and moved away from the scale model for four minutes to avoid any further heating from longwave radiation[8,9]. This sequence was repeated to simulate two days in the life of the shipping container, and to test the fidelity of the numerical solution scheme by comparison to the experimental results. The sequential heating experiment provided insight into how the model adjusted to new inputs with new initial T_F and T_S.

---

[8] Even though this amount of heat would be small, the heat transfer relation given in Equation 3 specifies that there is no external heating source after the time of insolation has been completed.

[9] The seemingly arbitrary time choice of 680s was a relic of a previous experimental measurement regime that relied upon very distinct heating times. However, even though that line of investigation was not pursued, the data that were collected were still useful in model validation.

Figure 12: The heating experiment in its four modes: (a) right side heating, (b) top side heating, (c) left side heating, and (d) "nighttime". The apparent light in the 12d is from a camera flash to make the setup visible.

Chapter 3: Results

In order to validate the numerical predictions of T_S and T_F, the model was tested under conditions that could be used as inputs to the model. These conditions were air temperature, albedo, initial T_S and T_F, and the dimensions of the model. Air temperature was measured using a DS18b20 sensor (Elenker, California). Albedo was approximated using the freely available program ImageJ and was calculated to be ~0.13[10]. The initial T_S value was measured before heating began using the IRT207. The initial T_F value was approximated using the average of the values measured by the probes before heating began. The dimensions were already measured and did not require reobservation, but they were important inputs to achieve the most similar numerical prediction. The initial condition for the first day simulation was $T_S = T_F = T_{Air} =$

---

[10] This approximation was done by following the method specified on albedodreams.info.

297K. Utilizing these conditions, T_S was calculated and plotted as a function of time for each of the three heated sides of the container and compared with the measurements made using the IRT207.

3.1: Model Validation Experiments
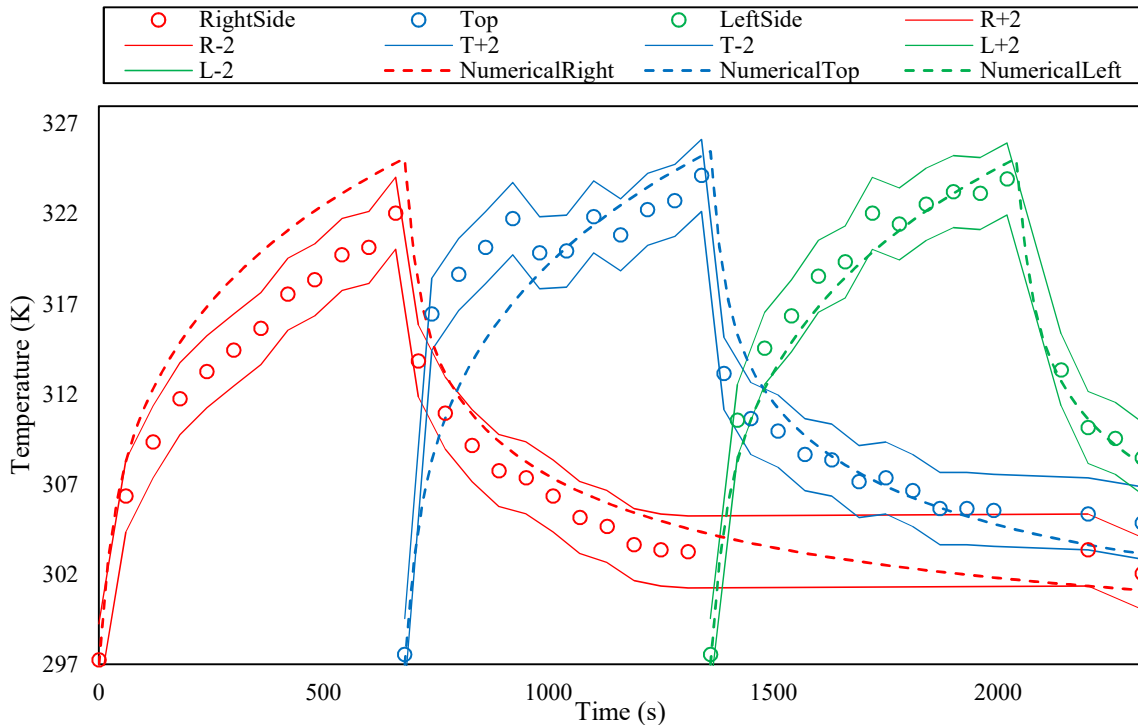
Figure 13: A plot of the numerically predicted and experimentally measured T_S after one "day" of heating. Solid lines represent the error bounds on the measured temperature.

Figure 13 shows the numerical predictions of the temperature of the steel against the measurements made during the heating experiment. The open circles in the plot represent the temperature of the steel (measured at the center of the wall with the IRT207), and the solid lines represent the uncertainty bounds on these measurements (i.e. $\mp 2°C$). The dashed lines are an interpolation of the discrete T_S values that are put out by MATLAB.

Utilizing these numerical outputs, the two-dimensional finite difference approximation was carried out to yield the following two-dimensional temperature distribution, $T(x,y,t)$.
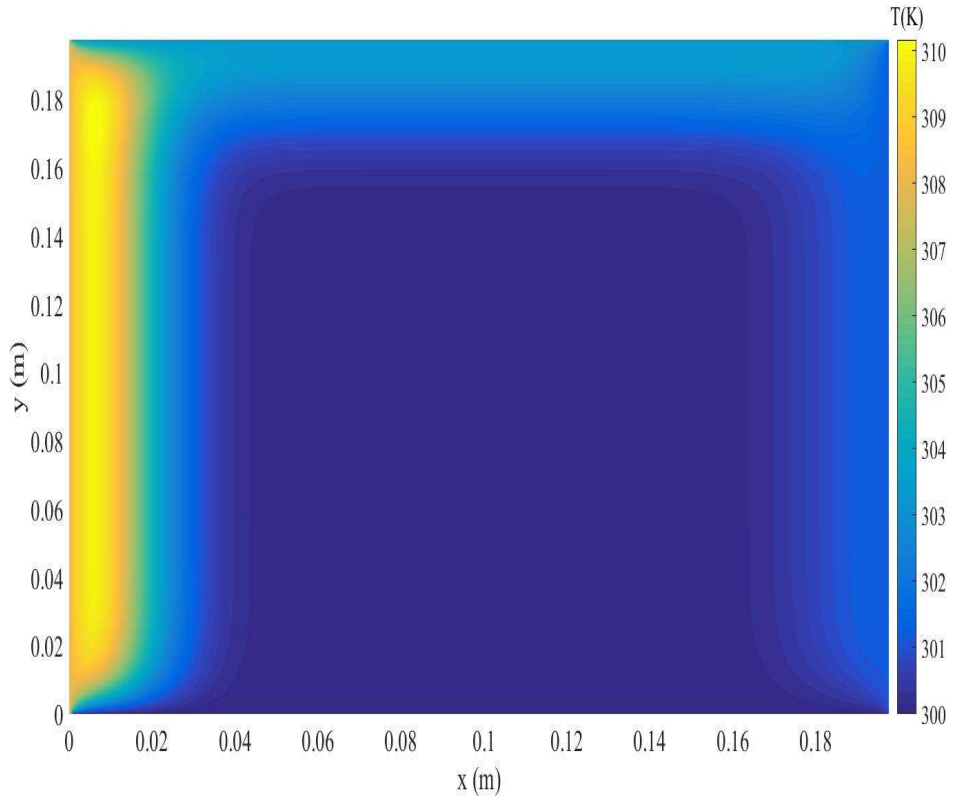
Figure 14: A heat map of the predicted temperatures within the scale model after one "day" of heating by shortwave radiation on each side.

This heat map can be compared to the discrete measurements made with the internal sensors by comparing two matrices: one populated with the numerical predictions and the other populated with the experimental measurements. Further, a comparison of these two matrices populated by the temperature rise of each point allows comparison of the heat transfer as expressed in Equation 7.

The matrices are designed to mimic the temperatures and the associated temperature rises (ΔT) of each discrete location within the continuum using Lord Kelvin's unit. The matrices are developed in the following manner:

$$\mathbf{T(x,y,t)} \equiv \begin{bmatrix} T(50cm,20cm,t) & T(100cm,20cm,t) & T(150cm,20cm,t) \\ T(50cm,60cm,t) & T(100cm,60cm,t) & T(150cm,60cm,t) \\ T(50cm,100cm,t) & T(100cm,100cm,t) & T(150cm,100cm,t) \\ T(50cm,140cm,t) & T(100cm,140cm,t) & T(150cm,140cm,t) \\ T(50cm,180cm,t) & T(100cm,180cm,t) & T(150cm,180cm,t) \end{bmatrix} \qquad (16)$$

In these matrices the x-coordinate defines the distance away from the left wall of the scale model and the y-coordinate defines the distance down from the top wall of the model. The time, t is a constant within each matrix as they represent a snapshot of the temperature distribution.

Table 2: Internal temperatures after one "day" of heating in units of K. (a) displays experimental measurements of internal temperature at the discrete grid locations; (b) displays predicted temperatures from the two-dimensional finite difference approximation at those same locations.

a / b

307.96	297.25	297.34	300.00	299.9	300.00
307.03	297.34	296.4	297.09	297.01	297.04
306.21	297.02	296.71	297.08	297.00	297.02
303.27	296.84	296.96	297.08	297.00	297.02
301.02	297.9	297.46	297.07	297.00	297.02

Table 3: Temperature rises after one "day" of heating in units of K. (a) displays experimental measurements of ΔT at the discrete grid locations; (b) displays predicted values of ΔT from the two-dimensional finite difference approximation at those same locations.

a / b

10.69	0.44	0.63	3.00	2.9	3.00
9.94	0.5	-0.06	0.09	0.01	0.04
9.37	0	0	0.08	0.00	0.02
6.31	0.63	0.31	0.08	0.00	0.02
4.18	1.13	0.94	0.07	0.00	0.02

The simulation of a second "day" of heating was carried out in the same way after the lamp had been removed from the left side of the box for four minutes.
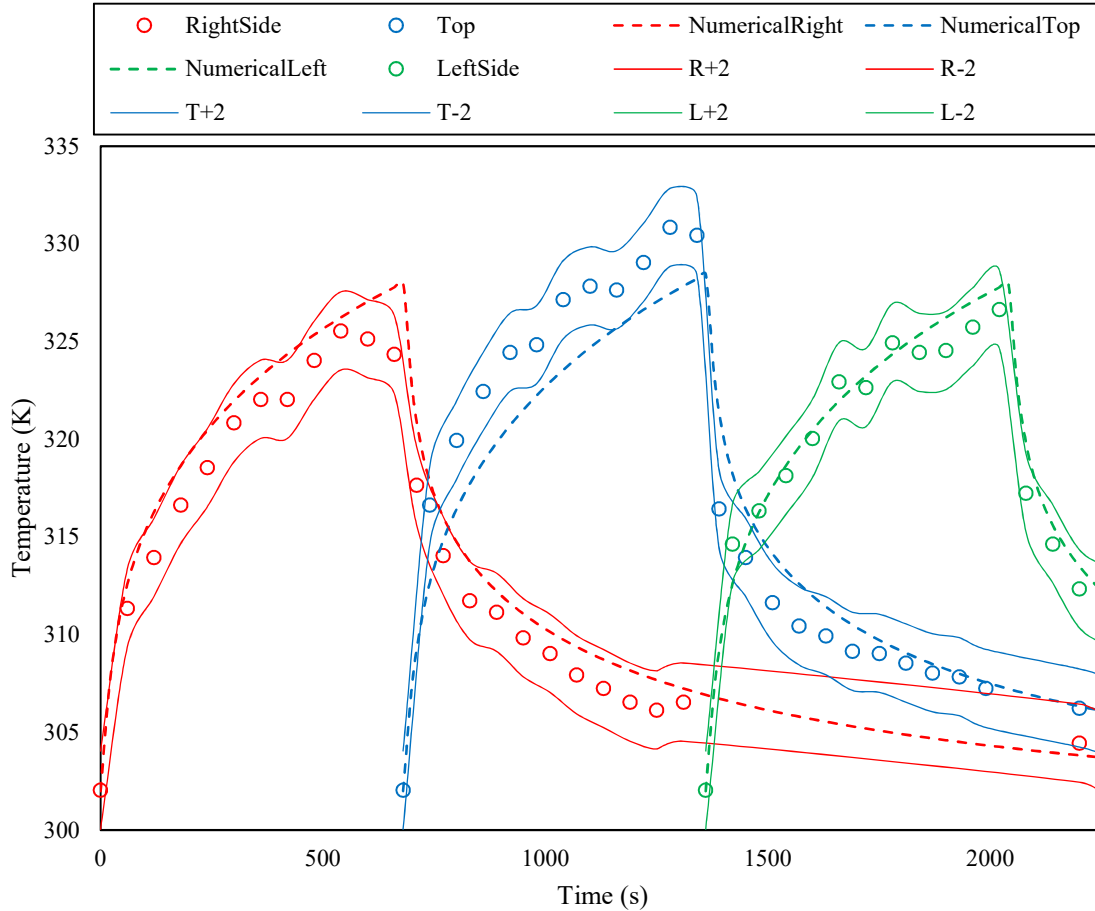
Figure 15: A plot of the numerically predicted and experimentally measured T_S after a second "day" of heating. Solid lines represent the error bounds on the measured temperature.

Again, the numerical outputs that made up T_S as a function of time were used as time dependent boundary conditions for the approximation of the conductive heating of the enclosed feces, and the results of this approximation are displayed below. However, in this simulation the initial temperature distribution was the ultimate output of the approximation from the first day of simulation. This method of making the outputs from the prior day the initial conditions for the subsequent day provides a method for utilizing this scheme to simulate the heating of the shipping container for periods of many days.

Table 4: Internal temperatures after a second "day" of heating in units of K. (a) displays experimental measurements of internal temperature at the discrete grid locations; (b) displays predicted temperatures from the two-dimensional finite difference approximation at those same locations.

a   **b**

310.71	299.5	299.27	302.89	302.38	302.49
309.15	299.96	296.9	298.05	297.2	297.41
308.34	297.21	297.09	297.87	297.002	297.22
305.02	298.65	297.59	297.87	297.001	297.22
303.02	299.59	298.9	297.6	297.0007	297.15

Table 5: Temperature rises after a second "day" of heating in units of K. (a) displays experimental measurements of ΔT at the discrete grid locations; (b) displays predicted values of ΔT from the two-dimensional finite difference approximation at those same locations.

a   **b**

4.25	2	1.75	2.89	2.48	2.49
3.25	2.31	0.44	0.96	0.19	0.37
3.19	0.19	0.32	0.79	0.002	0.2
2.56	1.69	0.69	0.79	0.001	0.2
2.37	0.69	1.31	0.53	0.0007	0.13
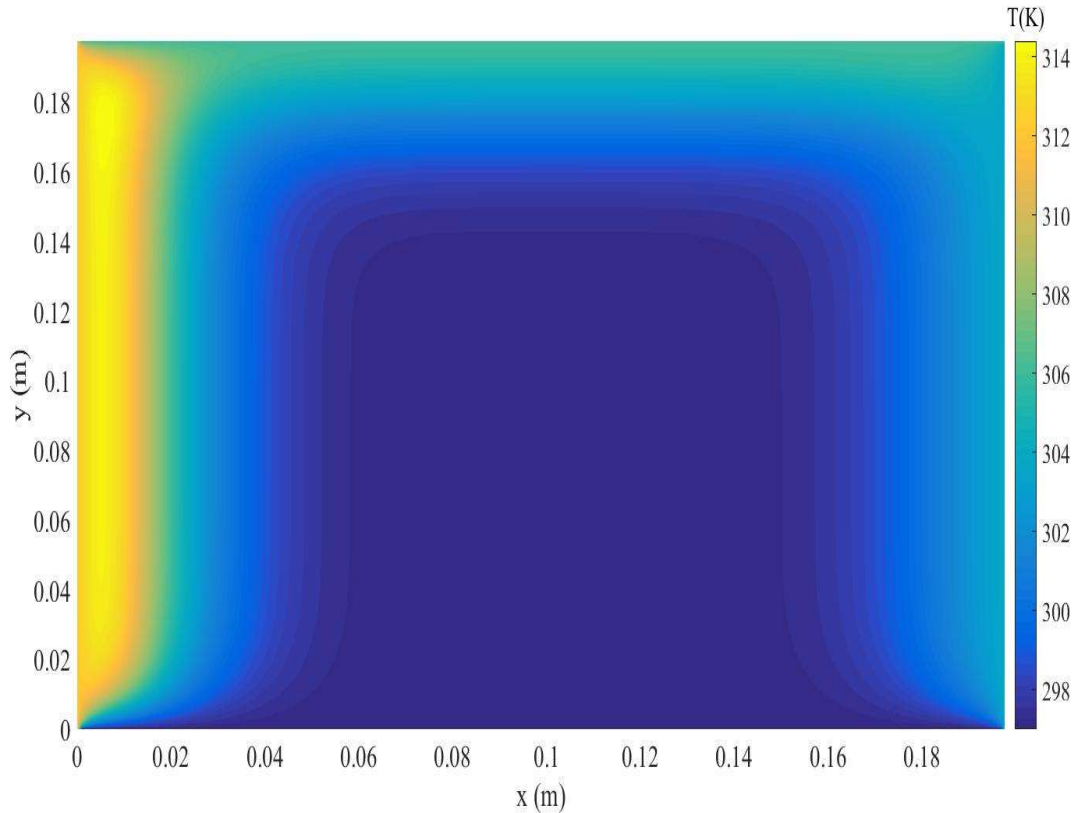
Figure 16: A heat map of the predicted temperatures within the scale model after a second "day" of heating by shortwave radiation on each side.

It is immediately apparent that the interior temperatures do not match the theoretical predictions, but there are several factors that may play into this discrepancy and reduce its significance. Firstly, the same pattern of interior heating is observed: the top and left are heated the most while the bottom and central portions of the continuum see the smallest temperature rise. This suggests that the heat is being transferred conductively, as the heating has the same qualitative effect on the continuum. However, there may be a discrepancy between the thermal conductance assigned to the fecal simulant (which is used in the model) and the actual thermal conductance of the enclosed simulant. This could perhaps be a result of the evaporation of water from the simulant, or other factors that could cause it to act more like a solid than a high viscosity fluid. Secondly, since the sensors were inserted into the model and it was moved many times after insertion, it is possible that they made measurements that were not exactly at the

31

points described by the sensor grid. Thirdly, since there were temperatures predicted in the model that are approximately equal to the measured temperatures, these outputs do not warrant immediate invalidation. If anything, these outputs, in comparison to the experimentally measured temperature distribution, show that the finite difference approximation using only conductive heat transfer is a conservative estimate of the heating that will occur.

With these results, it is possible to conclude that the model accurately predicts the thermodynamics of the proposed design. Therefore, the model can be applied at the large-scale in order to determine the feasibility of the proposed design, and to determine an approximate, necessary residence time for total inactivation. The agreement between the model and experimental measurements cannot be used to say that the large-scale container will behave in the exact same manner as the model. It would be necessary to perform a test of the large-scale design before implementing it in a community.

3.2: Prototype Predictions

In updating this program to make predictions about the proposed design prototype, the dimensions were updated to 2.6 m x 2.6 m x 6.0 m and the length of heating for each side was lengthened to 4 hours (14400 s). Further, spatial and temporal steps in the finite difference scheme were altered such that Von Neumann stability was maintained. The air temperature as well as the initial steel and feces temperatures were all changed to be 300K for ease of comparison.
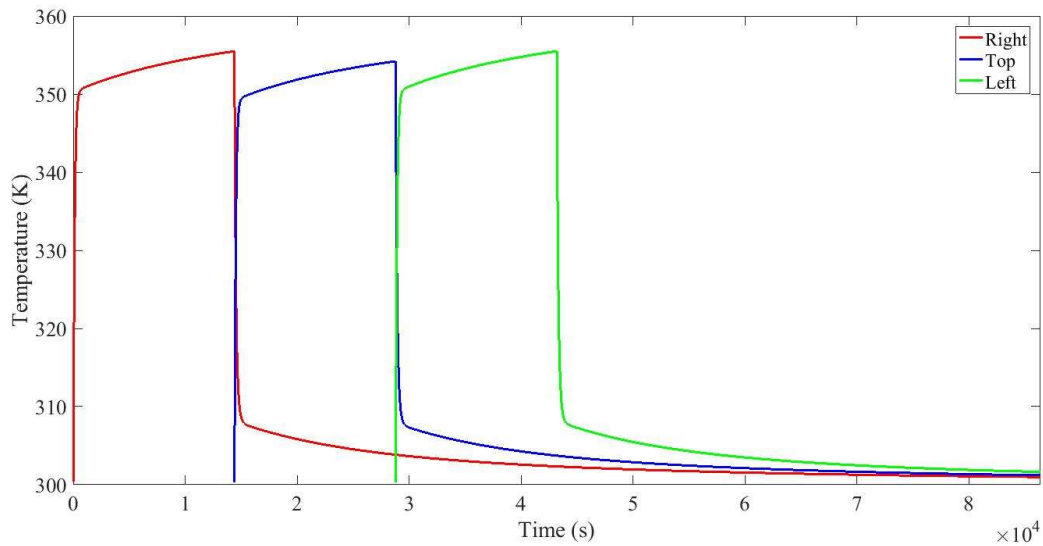
Figure 17: Temperature vs. time curves for the steel plates in the prototype design simulation with initial temperatures of 300K.

These predictions for $T_S(t)$ can be used as boundary conditions for the numerical solution of

Equation 13. A single day solution is presented below.
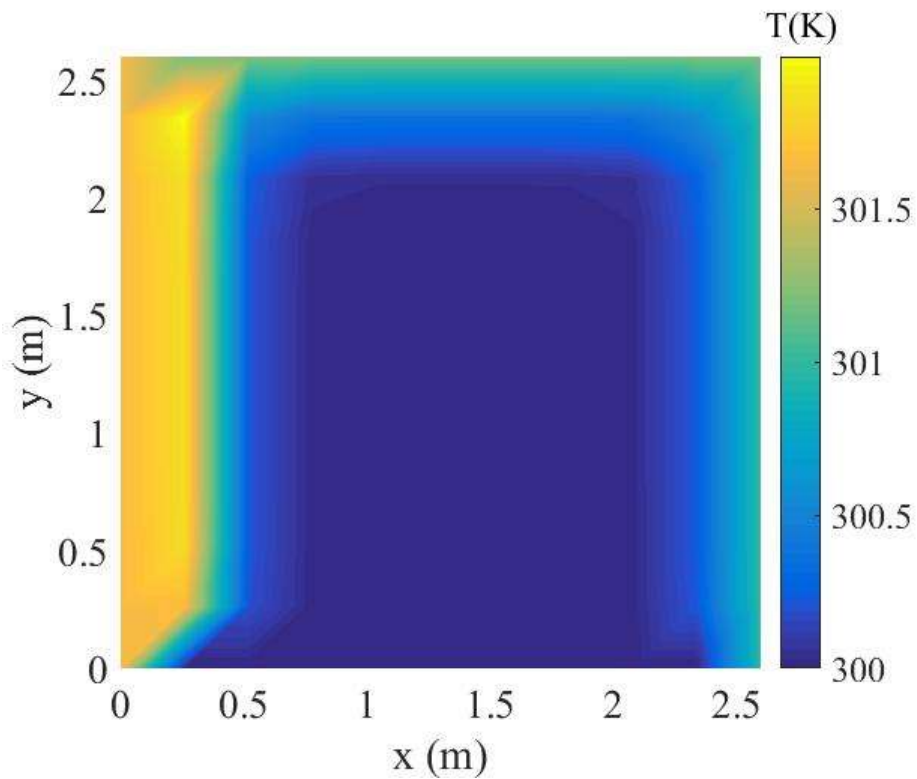
Figure 18: A heatmap of an approximated solution to Equation 13 after one day of heating.

These plots show that the size of the system does not effectively alter the physics. While the relative magnitudes of the temperature rise and the time scale for heat penetration may vary between the model and prototype calculations, the qualitative behavior of the outputs does not. Utilizing this same set of boundary conditions, the approximation was calculated to yield the temperature distribution after 100 days of heating. Calculations were also made for the case in which the container was rotated in the polar plane by 180 degrees after 100 days of heating. Additionally, calculations were made for various air temperatures, as they will naturally vary by region. These calculations are presented below.
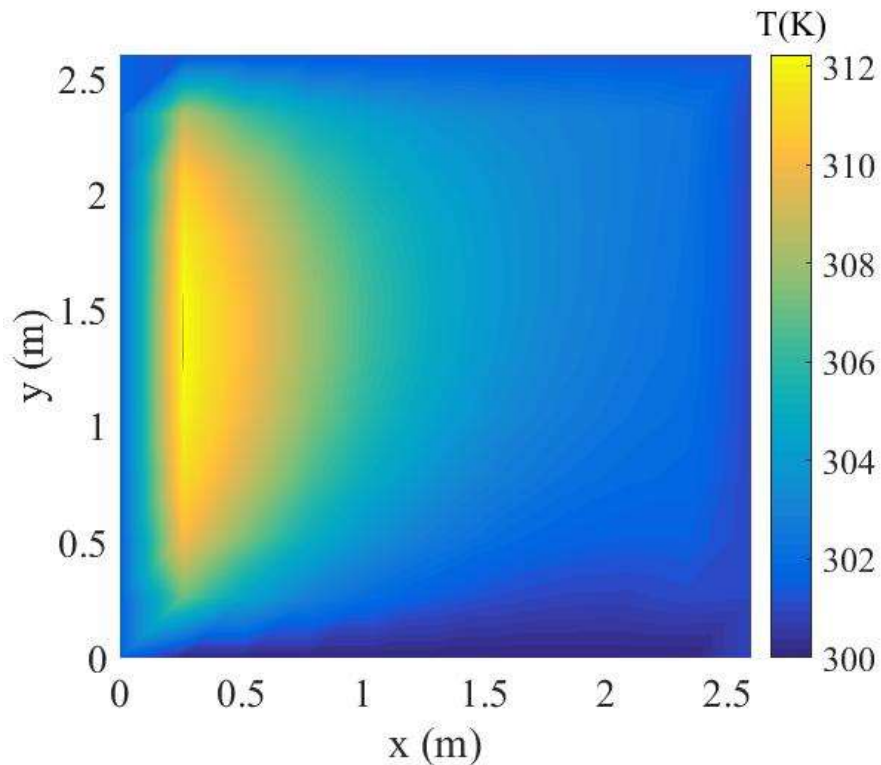
Figure 19: A heatmap of an approximated solution to Equation 13 after 100 days of heating with initial temperatures equal to 300K.
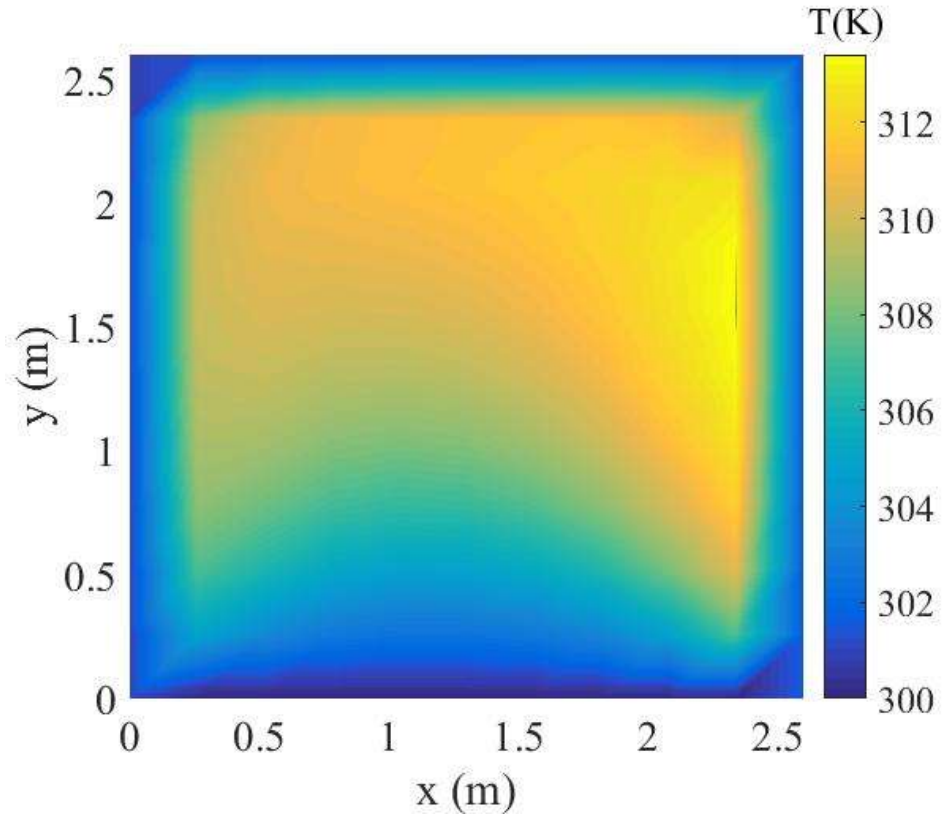
Figure 20: A heatmap of an approximated solution to Equation 13 after 200 days of heating with initial temperatures equal to 300K including a polar rotation of the container of 180 degrees after day 100 of heating.
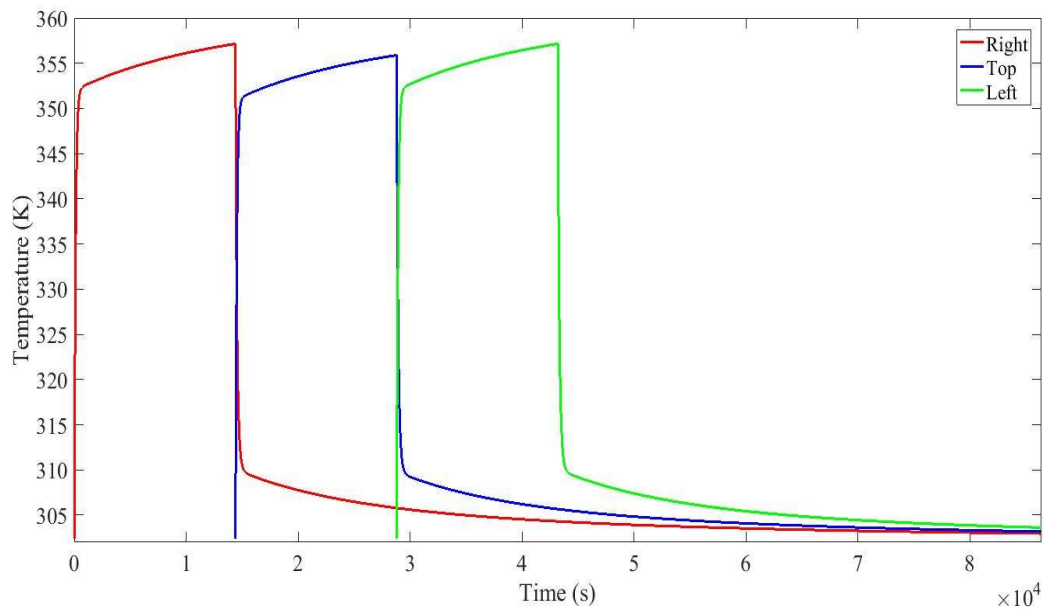
Figure 21: Temperature vs. time curves for the steel plates in the prototype design simulation with initial temperatures of 302K.
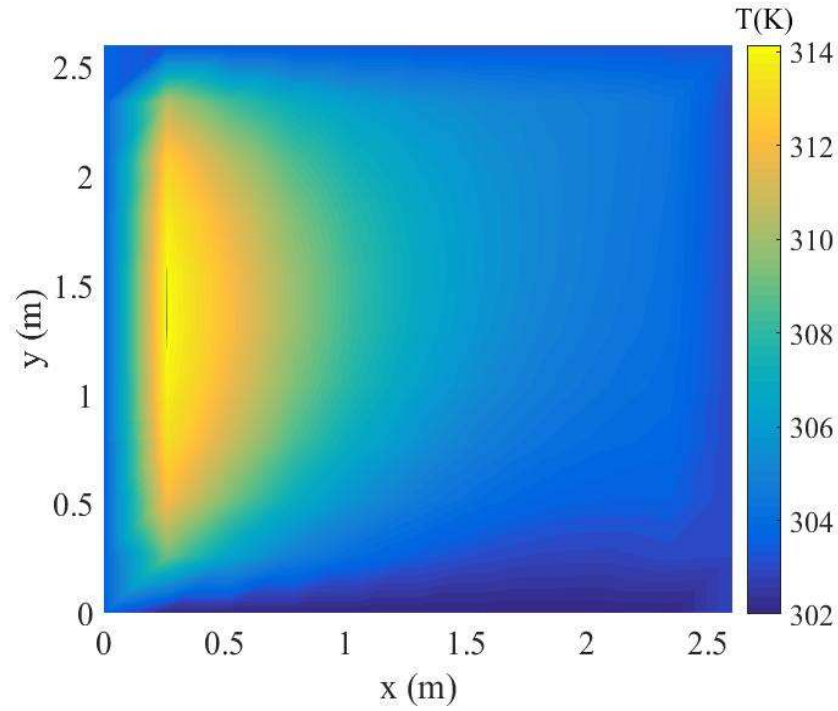
Figure 22: A heatmap of an approximated solution to Equation 13 after 100 days of heating with initial temperatures equal to 302 K.
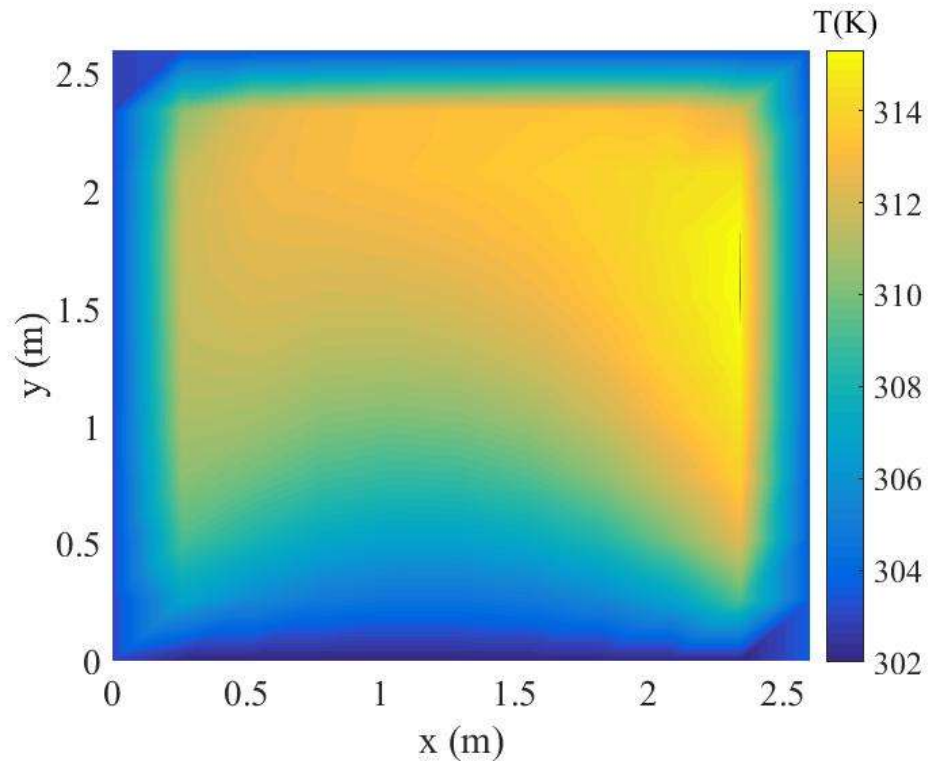
Figure 23: A heatmap of an approximated solution to Equation 13 after 200 days of heating with initial temperatures equal to 302K including a polar rotation of the container of 180 degrees after day 100 of heating.

Chapter 4: Discussion

These numerical outputs show that, without alterations, the proposed design will not reach temperatures necessary to completely inactivate all pathogens. There are some locations, namely those within 0.5 -1 m from the walls, that do reach the required temperatures for inactivation. But, by and large, the results show that solar heating alone is insufficient to heat an average shipping container to temperatures that will inactivate all fecal pathogens and microorganisms.

This is not the end of the potential for this design, however. Firstly, it must be noted that these numerical predictions, although the result of a validated numerical model, must be checked against a full-scale experiment. The lab-scale model was only used to verify this numerical model for the conceptual model presented above. Further, certain quantities like the thermal conductance of the enclosed feces and the actual albedo of the surface of the shipping container must be considered. In addition, the effects of the geometry of an actual shipping container (e.g., corrugated walls) may impact the heating of the enclosed feces. Additionally, constant air temperatures and a lack of forced convection (i.e., wind) were unrealistic but necessary components of this conceptual model. Furthermore, this numerical simulation begs for a full-scale test of the proposed design since the achieved internal temperatures were very close to, and in some locations met, those required to achieve inactivation. Also, since higher internal temperatures were observed within the scale model during experimentation it may be possible that in the case of a full-scale test the same exaggerated heating may be observed.

Additionally, it must be stated that a full-scale test would need to be done with this design before implementation to measure the effectiveness of the thermal inactivation.

Specifically, the fecal continuum would need to be sampled and analyzed for pathogens before and after heating was completed. This is necessary since the requisite time and temperature for total inactivation at low temperatures is not firmly agreed upon.

Also, the potential for biogas production in this design would need to be studied. Since the predicted temperatures for much of the design hover around 310K (~37°C), anaerobic digestion, and thus methanogenesis, is likely to occur and be very productive (Colón et al., 2015). This methane generation has two important consequences for the results of this experiment. Firstly, while the design does not show success with respect to rapid inactivation of pathogens, there is the potential for success with respect to biogas generation. This biogas could ultimately be used to help support the energy demands of developing communities. Secondly, the methane production, and its subsequent removal, would require the use of degassing devices (Colón et al., 2015). The insertion of degassing equipment would alter the heat transfer model proposed above as there would be fluid (e.g., gaseous methane) moving through the continuum, thus adding an internal convective heat transfer term to the conceptual model. The effect of this convection would require further study at an experimental scale to determine its effect on the heating of the feces at large. In addition, the biogas generated by this system would likely require significant treatment and purification as other gas-yielding reactions will yield a biogas of more components than just methane (Rose et al., 2015).

Another point to be considered with these results is the lack of mention of the effect of urine as well as cleansing products (e.g., toilet paper) upon the potential for anaerobic digestion and the thermal properties of the enclosed sludge. The separation of feces and urine has been shown to behoove methane production (Colón et al., 2015; Rose et al., 2015). Also, thermal conductance of the fecal sludge is likely to change with the combination of urine and cleansing

products with the feces, as can be seen by the difference between the thermal conductivity of fresh feces versus that of fecal sludge (Pollution Research Group, 2014). Further, the thermal conductance of feces has been shown to increase with volumetric water content (Iwabuchi et al., 1999), so it would be important to limit excessive drying prior to storage in the container.

4.1: Proposed Design Enhancements

Since the numerical predictions are near the range required to inactivate fecal pathogens, certain design changes could be made in order to expediate and/or enhance the heating of the enclosed feces. These design alterations, which could be implemented separately or in conjunction with one and other, would require testing at the large-scale since they are difficult to model numerically. The proposed alterations to the design are: nighttime insulation, permanent insulation of the bottom plate, addition of reflective surfaces to increase incident radiation intensity, addition of air supplying pipes parallel with the longitudinal axis of the container, and the insertion of heating elements parallel with the longitudinal axis of the container.

It has been observed in experimental studies that "sufficient insulation" is necessary for the effectiveness of thermal composting in almost all environments (Vinnerås et al., 2003; Niwagaba et al., 2003). In pilot studies by Vinnerås and Niwagaba (2003), the goal was to ensure that the heat, which moved from the center of the composting reactor outwards, did not simply conduct through the external regions, but effectively heated the feces to inactivation temperatures. In this design however, the goal is to ensure that heat is directed more internally than externally. This could be assisted by placing an insulated cover over the shipping container just before the sun sets so that the hot, left wall is maintained at a temperature near to its maximum for a longer period, thus lengthening the heating of the feces.

A common denominator in both the experimental study of the lab-scale model and the numerical approximation was the bottom plate of the container acting as a heat sink. Across the board, lower temperatures were observed towards the bottom of the enclosure. This is partially due to the temperature gradient caused by the natural convective heat loss on the exterior of the container, but it is also due to heat being transferred through the bottom plate into the ground. This problem could be circumvented by adding a layer of insulating foam to the inside of the bottom of the container such that conductive heat flux to the ground would be greatly reduced. These two design alterations would likely benefit the heating of the feces and cause greater heating of the feces while requiring only a very small addition of labor and resources. Another way to expedite the heating of the enclosed feces would be to add reflective surfaces adjacent to the container to increase the intensity of the radiation incident upon the container. Adding these reflective surfaces would be low in cost and would increase S in Equation 2 because the shortwave radiation reaching the container would be composed of directly and indirectly (i.e., reflected) incident components. The next two potential solutions, however, are much more fiscally, temporally, and energetically taxing.

The composting of feces has been widely studied in both the academic (Vinnerås et al., 2003; Niwagaba et al., 2003) and agricultural (Jenkins, 2005) fields. These studies and practices have largely been of aerobic composters, whereas the design proposed in this study is anaerobic. Aerobic composting processes frequently achieve temperatures of $70\,^\circ$C and are consistently over $50\,^\circ$C. Therefore, it would be desirable to utilize the biogenic heating that can be developed under aerobic conditions. This would require that aerobic conditions be supplied to the design. One could accomplish this by inserting long, narrow, perforated pipes into the shipping container parallel to its longitudinal axis and pulsing air through them on a regular basis. Experiments

would have to be done to determine the optimal number and location of the pipes, as well as the required airflow to encourage aerobic respiration within the enclosure. Further, it would be immensely important to ensure that urine was diverted to a separate collecting vessel so that the carbon-to-nitrogen ratio within the enclosed feces was managed to encourage the composting process (Niwagaba et al., 2003). This alteration is not desirable because it relaxes the requirements that this design be low-cost and low maintenance. Further, the pipes would need to be supplied with air from a pump that would need to be powered, and the pipes would require cleaning out if the perforations were to become occluded with feces.

Finally, treatment could be enhanced by the placement of heating elements throughout the fecal continuum[11]. These point sources of heating would allow heat to reach locations within the closed container to which heat had not penetrated via conduction. This would likely allow for quicker and more even heating of the feces. The heating units could be powered by an onsite solar cell. The drawbacks to this alteration are the difficulty of implementation (the heating elements would need to be placed inside the container while it was being filled with feces), the initial cost to purchase the heating elements, and their constant power draw. Even though the power requirement could likely be met by the use of a solar cell, it would likely be preferable to use the electrical energy generated in a developing community toward other ends.

Chapter 4.2: Insulation Modeling

Chapter 4.2.1: Approximation using an Insulated Boundary Condition

Adding insulation to the bottom and end caps of the container as well as placing an insulating cover over the hot walls at the end of the day are the simplest design alterations presented in this thesis. Therefore, it is pertinent to estimate their effect on the heating of the

---

[11] This solution was at the suggestion of Mario Montesdeoca

41

proposed design. Insulation was initially treated as a first-order approximation of a no heat flux

boundary condition where the boundary temperature was set equal to that of the adjacent node

(Dai, 2010). For the bottom wall of the container, this condition was maintained for the entirety

of the simulation. For the exposed walls of the container, the no-flux condition was imposed

after 12 hours of irradiance had passed. Under these conditions, the enclosed feces reaches a

temperature no less than 46˚C (i.e., 319K) after 136 days. Concurrent with the previously

completed simulations, these calculations account for a 180˚rotation midway through the

modeled implementation time. The result of this simulation can be seen in Figure 24.
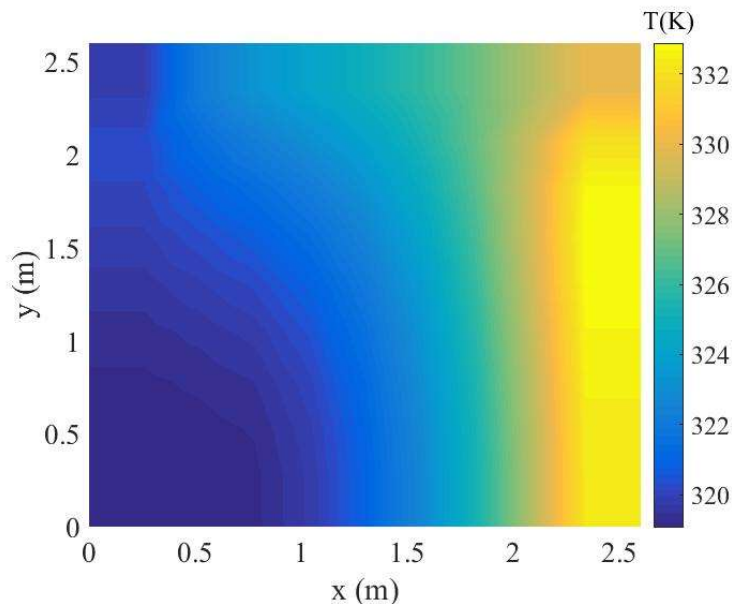
Figure 24: A heatmap of an approximated solution to Equation 13 after 136 days of heating with
initial temperatures equal to 300K including a polar rotation of the container of 180 degrees after
day 68 of heating. This container has its bottom panel insulated for the entirety of the simulation
and the hot walls insulated after 12 daily hours of irradiance. Simulation of the insulated panels
is done by equating boundary temperatures to those of the adjacent nodes.

Implementing the no-flux boundary condition by setting the boundary temperature equal

to that of the adjacent node is only a first order approximation of the physics, thus it is desirable

to make a more robust simulation of insulated walls (Dai, 2010). This can be done by setting the

temperature gradient across the boundary equal to zero. Computationally, this is done by creating

a set of "fictitious" nodes outside the domain that mirror the set of nodes immediately adjacent to the boundary, yielding a zero temperature gradient across the boundary (Dai, 2010).

Simulation of the insulated walls with the second order approximation yielded slightly different results. Various orientations of the container were modeled in this simulation. Since the wall that is irradiated last will be the hottest during the nighttime period of insulation, the feces nearest to that wall will be hotter than the rest of the enclosed feces. Three 180° polar rotations each separated by a period of forty days yielded the temperature distributions in Figure 25 that correspond to the temperatures after 40,80,120, and 160 days, respectively.
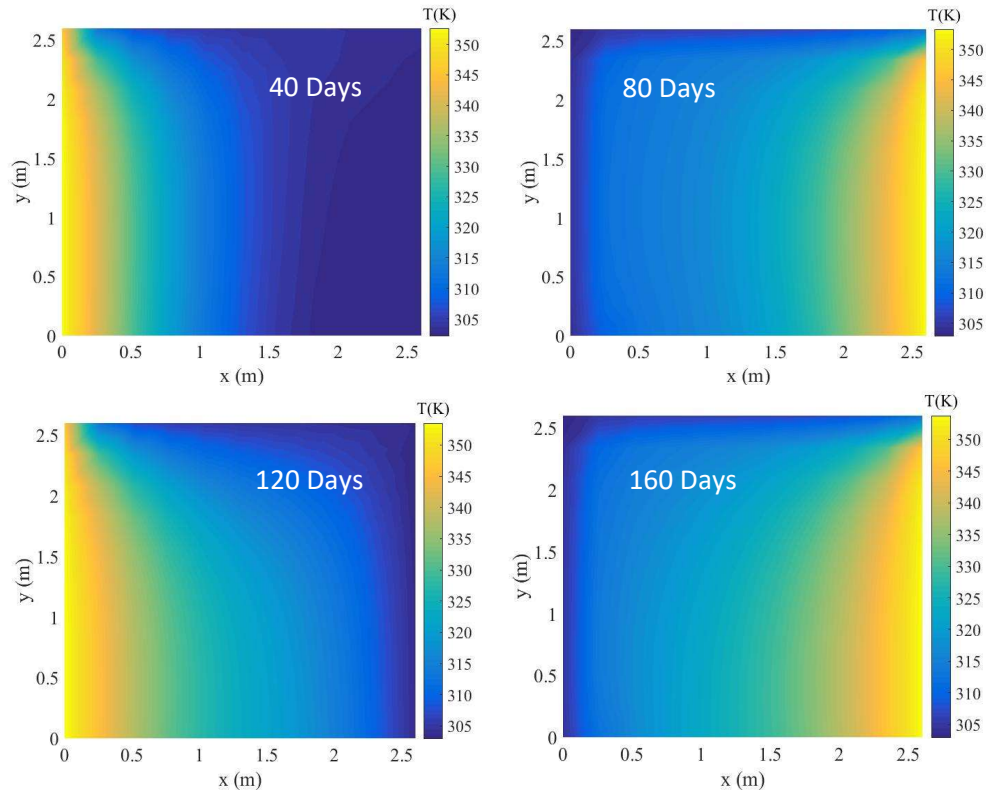
Figure 25: Heatmaps of approximated solutions to Equation 13 after 160 days of heating with initial temperatures equal to 300K including a polar rotation of the container by 180 degrees after each successive 40-day period of heating. This container has its bottom panel insulated for the entirety of the simulation and the hot walls insulated after 12 daily hours of irradiance. Simulation of the insulated panels is done by eliminating transboundary temperature gradients. Each image corresponds to the heatmap after 40 subsequent days have passed.

These predictions show that the updated implementation of the insulated boundary condition

does not predict the same complete exceedance of 319K as the first order approximation. But,

except for the top few rows of cells, these data show that temperatures of 319K can be reached

and exceeded during the field implementation of the prototype. This can be seen in Figure 26,

which shows the cells that reach 319K after the second or third reorientation of the container.
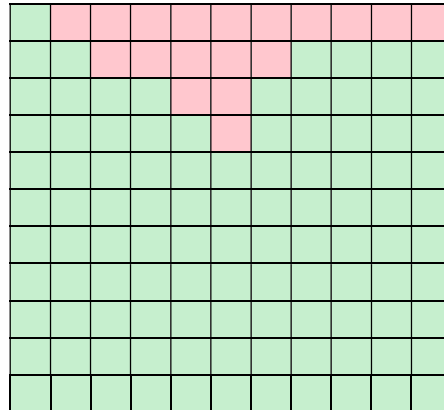
Figure 26: This figure shows the nodes (i.e. $T_{i,j}$) within the enclosed feces that reach or exceed 319K after the second or third reorientation of the container. These nodes are highlighted in green. Nodes that do not reach this temperature requirement are highlighted in red.

The top row of nodes is not a concern in the consideration of pathogen inactivation because the

hot wall achieves temperatures that promote extremely rapid inactivation (Feachem et al., 1983).

If one factors in the midday temperature fluctuations, then certain nodes that are filled with red

in Figure 26 actually reach the required 319K; this result is shown in Figure 27.
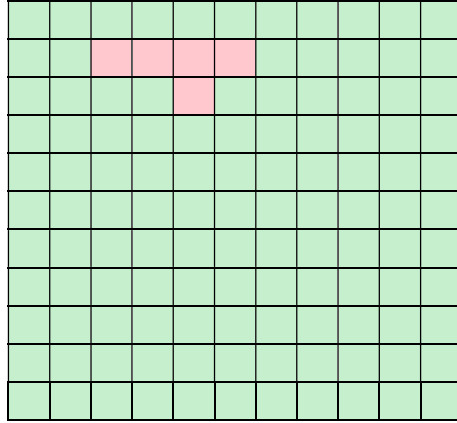
Figure 27: This figure shows the nodes (i.e. $T_{i,j}$) within the enclosed feces that reach or exceed 319K after the second or third reorientation of the container, as well as those that reach 319K during the middle of the irradiation period. These nodes are highlighted in green. Nodes that do not reach this temperature requirement are highlighted in red.

This result shows that there are still some nodes within the enclosed feces that are not predicted to reach 319K. Most of the predicted temperatures within the shipping container meet or exceed the chosen temperature requirement of 319K.

The outputs of the updated models presented in this section provide further support for the need to test this design at the prototype scale. The first and second order approximations show that there is the potential for the temperature of the enclosed feces to reach or exceed 319K.

Chapter 4.2.2: Approximation using Various R-values

Insulation can also be simulated using the energy balance method introduced in chapter 2. Again, the bottom plate of the container is permanently insulated. This simulation considers that each wall of the container is insulated immediately after it receives four hours of normally incident shortwave radiation at an intensity of 1000 W/m^2. Polystyrene is used as the simulated insulator in this investigation because it is cheap and ubiquitous. The insulating potential of a material is described by its R value (given in Equation 15), the ratio of the material thickness to

its thermal conductivity (Hagen, 1999). Therefore, a larger R value is indicative of a better insulator.

$$R = \frac{\Delta x}{k} \tag{15}$$

Utilizing this ratio, a new heat transfer relation can be written for the nighttime cooling of the steel wall:

$$\dot{q} = -k_F \frac{(T_S - T_F)}{\delta x} - k_p \frac{(T_S - T_A)}{\Delta x} = -k_F \frac{(T_S - T_F)}{\delta x} - \frac{(T_S - T_A)}{R_p} \tag{16}$$

Where:

$$k_p \equiv \text{Thermal conductivity of polystyrene} = 0.029 \ \frac{W}{m \cdot K} \tag{17a}$$

$$R_p \equiv \text{R value of polystyrene} \tag{17b}$$

Equation 16 is used to calculate the temporal evolution of the steel wall temperature under the assumptions that conduction is the most significant heat loss mechanism. It replaces Equation 3 in the iterative integration scheme. It says that heat will be lost conductively to the enclosed feces and surrounding air in proportion with the associated temperature differences and thermal conductivities. Two different simulations were run that each correspond to two different R values for the installed insulation: 0.85 K·m^2/W and 1.70 K·m^2/W. These values correspond to one and two inches of polystyrene insulation, respectively. It is assumed that convective and radiative losses are minimal for the insulated container.

The iterative integration and finite difference scheme laid out in chapter 2 was utilized to predict the temperature distribution within the enclosed feces as a function of time. Figures 28 and 29 reflect the simulation using R = 0.85 K·m^2/W, and Figures 30 and 31 reflect the simulation using R = 1.70 K·m^2/W.
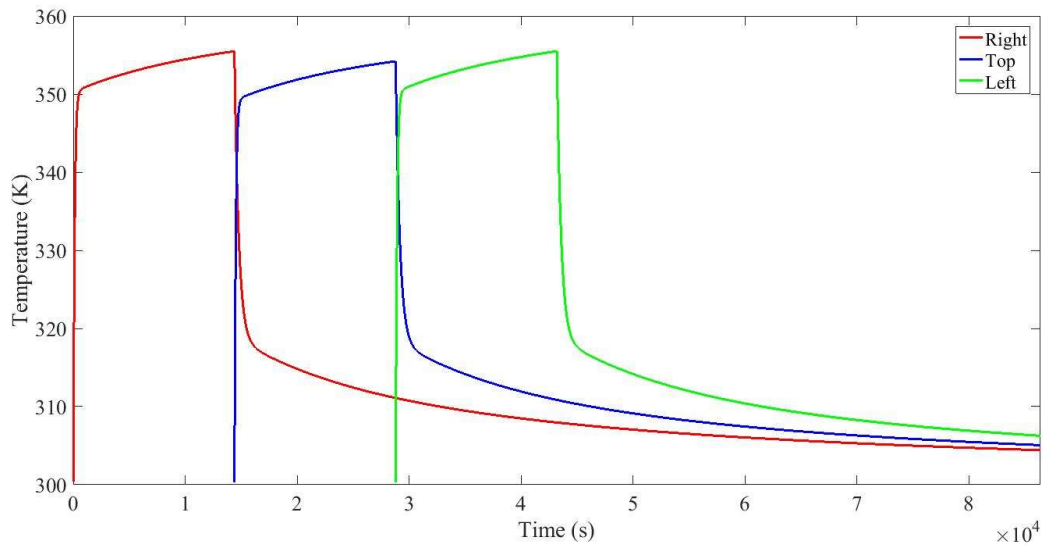
Figure 28: Temperature vs. time curves for the steel plates for the prototype design simulation with initial temperatures of 300K and nighttime insulation. The insulation used in the simulation is polystyrene with an R value of 0.85 K·m^2/W.
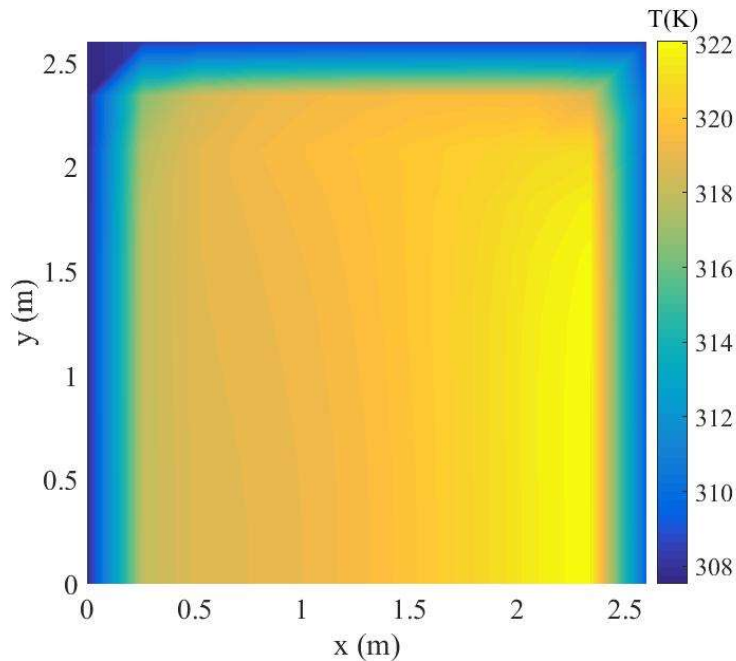
Figure 29: A heatmap of an approximated solution to Equation 13 after 171 days of heating with initial temperatures equal to 300K including a polar rotation of the container of 180 degrees after day 85 of heating. This container has its bottom panel insulated for the entirety of the simulation and the hot walls insulated after each receives 4 daily hours of irradiance. Simulation of the insulated panels is done using the iterative integration scheme from chapter 2. An R value of 0.85 K·m^2/W is used.
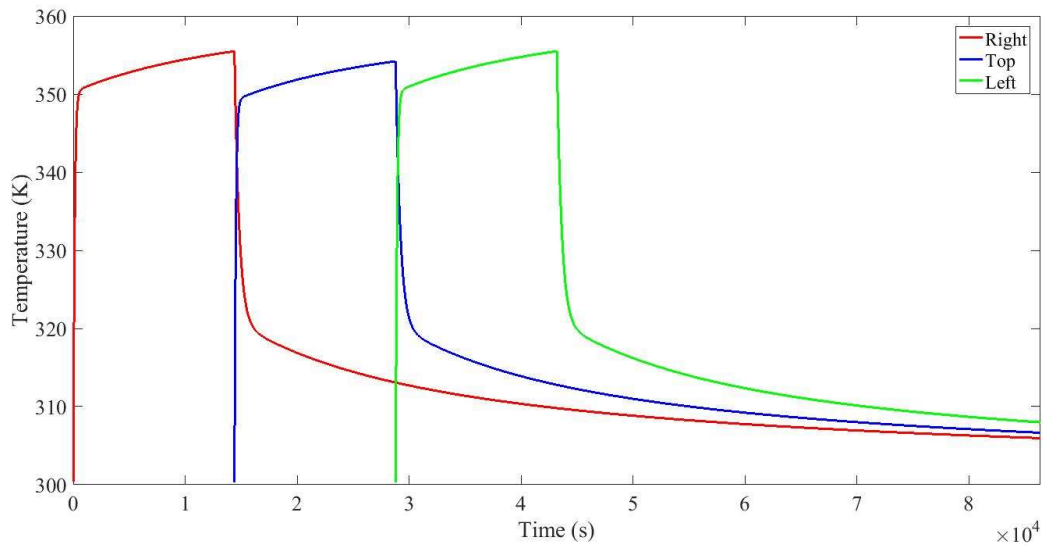
Figure 30: Temperature vs. time curves for the steel plates for the prototype design simulation with initial temperatures of 300K and nighttime insulation. The insulation used in the simulation is polystyrene with an R value of 1.70 K·m^2/W.
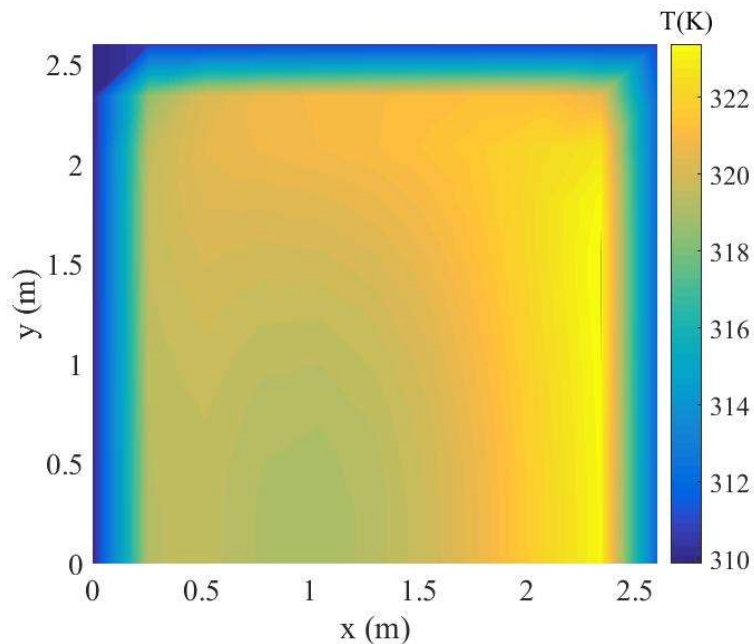
Figure 31: A heatmap of an approximated solution to Equation 13 after 117 days of heating with initial temperatures equal to 300K including a polar rotation of the container of 180 degrees after day 58 of heating. This container has its bottom panel insulated for the entirety of the simulation and the hot walls insulated after each receives 4 daily hours of irradiance. Simulation of the insulated panels is done using the iterative integration scheme from chapter 2. An R value of 1.70 K·m^2/W is used.

Figures 29 and 31 show the temperature distribution within the shipping container after 171 and 117 days, respectively. These time intervals are the time it takes for each of the systems to totally reach 319K or reach a steady state at every node. Insulation with an R value of 0.85 K·m^2/W did not yield temperatures greater than or equal to 319K at every node, but the predicted temperatures were predicted to be greater than or equal to 317K. Because the R value used to produce Figure 31 was larger, it was able to meet the temperature requirement. Utilizing an R value to alter the heat transfer relation, it was predicted that necessary temperatures for pathogenic inactivation would be reached after at least 117 days with 2 inches of polystyrene insulation. The implementation of the immediate insulation regime would marginally increase the labor requirements of the design, but this would allow for quicker heating and treatment of the manure. The two methods of analysis used to model the effect of insulation on the proposed design show that sufficient temperatures for inactivation are predicted to be reached after a treatment time of approximately 117 to 136 days, optimistically.

Chapter 5: Future Work

Since the predictions made by the numerical model are near the range at which pathogenic inactivation will occur, these results suggest the need for a full-scale experiment. This experiment should be completed in a laboratory space with fecal simulant (note: water or food waste may be better fecal simulants for a full-scale test because producing enough UKZN fecal simulant to fill a shipping container may prove difficult) and in a practical environment, using human or animal feces, at lower latitudes to validate the assumptions of solar radiation, air temperature, etc. Further, in a full-scale experiment it would be important to outfit the prototype with proper insulation and a nighttime cover. It may also behoove the development of this

prototype to consider the implementation of aeration tubes or heating elements *in situ*; however, these alterations significantly relax the low-cost and low-maintenance design requirements.

Additionally, three areas that require further study are the generation of biogas and pressure within the enclosure, maintaining the isolation of the enclosed feces and the structural stability of the prototype, and studying the inactivation kinetics of the design. The full-scale test should be instrumented with pressure sensors throughout the feces in order to understand the extent to which the gas that is generated can affect the structural integrity of the container. It should also be outfitted with the equipment required to degas the container to understand how this would affect the achieved temperature distribution. Since the goal of the design is to isolate human manure from a given community to limit the fecal pathogen vector, it would be important to ensure that the proposed design incorporates measures that prevent leakages from the liner and/or the container. This would include protection of the container from overland flow during high intensity rainfall events, and it would require protection of the surrounding environmental media from vessel leakages. The former could be dealt with by elevating the container and the latter could be dealt with by installing a concrete pad with a subsurface collection tank. Additionally, care would need to be taken in the placement of the vessel so that the potentially malodorous air surrounding it does not lessen the quality of life in the nearby community. The design of these or other safeguards warrants further research. Additionally, since polar rotations of the container are an important aspect of the model, installing a way to easily rotate the container would enhance treatment. Finally, since this analysis only used the achieved temperatures as an indicator of design feasibility, it is of the utmost importance to develop a methodology for understanding the inactivation kinetics of the design. Implementation of this treatment process outside of a research site before full inactivation can be verified is

discouraged. A rigorous method for sampling and analyzing the feces for sufficient inactivation of pathogens would need to be designed and tested before implementation could be carried out. Furthermore, such a biological method would need to be simple enough that it could be carried out during the lifetime of the prototype if it were to be implemented at the community level. In the case that this would not be possible, the investigation and determination of a suitable factor of safety to adjust the required treatment time would be a salient topic of further research.

Chapter 6: Conclusions

This study was conducted to determine the extent to which storing human feces in a lined shipping container that is heated primarily by shortwave solar radiation would yield inactivation of fecal pathogens. Using a physics-based model that was validated with a laboratory-scale model of a lined shipping container, it was predicted that the achieved temperature distribution within the container comes close to, but does not meet, the temperature requirement for pathogenic inactivation as is. Design alterations including insulation, aeration, and electrical heating elements were proposed to bolster the analyzed design. The effects of various insulation regimes were modelled, and it was predicted that temperatures no less than 319K could be achieved after a heating period on the range of 117 to 136 days. A full-scale test of the prototype is recommended to validate the scaling up of the conceptual model and to verify the accuracy of the predicted temperature distribution.

Chapter 7: Appendices

Appendix 1: Calculating Convective Heat Transfer Coefficients

The natural convective heat transfer coefficient, h_{NC}, was calculated using theoretical and empirical relations provided in Kirk Hagen's *Heat Transfer with Applications*. The relations vary with the orientation of the plate relative to the direction of gravitational acceleration since natural convective heat transfer results from a buoyancy driven flow. The general process for calculating h_{NC} is laid out below.

1: Calculate the Grashof number (Gr)

$$Gr = \frac{g\beta(T_s - T_A)L_C^3}{\nu_A^2} \qquad (A - 1.1)$$

Where:

$$g \equiv \text{Acceleration due to gravity} = 9.8\frac{m}{s^2} \qquad (A - 1.2a)$$

$$\beta \equiv \text{Thermal Volumetric Expansion Coefficient} = \frac{1}{T_A} \qquad (A - 1.2b)$$

$$L_C \equiv \text{Characteristic Length for convection} \qquad (A - 1.2c)$$

$$\nu_A \equiv \text{Kinematic Viscosity of Air} = 0.000015\frac{m^2}{s} \qquad (A - 1.2d)$$

2: Calculate the Rayleigh number (Ra)

$$Ra = Gr \cdot Pr_A \qquad (A - 1.3)$$

Where:

$$Pr_A \equiv \text{Prandtl number for Air} = 0.71 \qquad (A - 1.4)$$

3: Calculate the Nusselt number (Nu)

For vertical plates (i.e. side walls of the container):

$$Nu = \left\{ 0.825 + \frac{0.387Ra^{\frac{1}{6}}}{\left[1 + \left(\frac{0.492}{Pr_A}\right)^{\frac{9}{16}}\right]^{\frac{8}{27}}} \right\}^2 \qquad (A - 1.5a)$$

For horizontal plates (i.e. top wall of the container):

$$Nu = 0.15Ra^{\frac{1}{3}} \qquad (A - 1.5b)$$

4: Calculate h_{NC}:

$$h_{NC} = \frac{Nu \cdot k_A}{L_C} \qquad (A - 1.6)$$

Where:

$$k_A \equiv \text{Thermal conductivity of Air} = 0.028\frac{W}{m \cdot K} \qquad (A - 1.7)$$

Appendix 2: MATLAB Scripts used to Perform Model Calculations

The scripts used to make the calculations for each prediction provided in the results and discussion section of this thesis are presented below. These scripts can be run in MATLAB by copying and pasting them into the "Editor" and clicking run. They do not require inputs because they are scripts. Any variables (e.g. shortwave radiation intensity, ambient air temperature) can be adjusted within the script itself. The scripts used to make each set of calculations come in pairs. The first is used to determine the time-temperature curve for the steel (i.e. the boundary condition) and the second is used to calculate the spatiotemporal temperature distribution for a latitudinal slice of the container at its midpoint. They are to be used subsequently. Rotations are simulated by switching the side wall heating curves during the simulation. These rotations can be incorporated into the T(x,y,t) codes by adding copies of the loops to the code that has already been written.

Scale Model Calculations:
Calculating T(t):

```
%This code is used to calculate the rise and fall of the temperatures
%for the scale model experiment
%All times are presented in seconds


TT = 300000;


W = 0.1976;    %scale steel box width (m)
dx = W/50;     %Spatial discretization for conductive component
H = 0.1976;    %scale steel box height (m)
L = 0.456;      %scale steel box length (m)
wh = W/2;   %half the box width (m)
Ar = L*H;    %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;   %insolation (W/m2)
a = 0.1318;   %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.87; %steel emissivity (-)
Tao = 297; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.5;   %thermal conductivity of feces (W/mK)
tC = 0.000635; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tf = 300; %Feces temperature (K)


pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)


mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)


cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)
```

```matlab
A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;    %reciprocal of D


S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/(2*dx);    %fecal conductance/half container thickness (W/m2K)



alpha = kf/(pF * cF); %Define thermal diffusivity for feces

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

%initialize time and temperature arrays for output storage
timeS = zeros(1,1);
TempS = zeros(1,1);


tiSi = 0;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the right steel wall with time
    if tiSi > 680 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts+0.0003,'RelTol',1.6,'AbsTol',1.6); %how long
does it take for the steel to increase temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));                    %recalculate ho
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;
```

```matlab
    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end


    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 0:1:680;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));

Ts = TS1(1,681);
tiSi = 680;

for i = 1:TT %calculate temperature drop of the right steel wall with time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003,'RelTol',1.6,'AbsTol',1.6); %how long
does it take for the steel to decrease temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 minus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));        %recalculate ho
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 302;
    dt = dt/100;
```

```matlab
        Z = ((alpha) * (dt))/(dx ^ 2);
            if Z > 0.5
                disp('unstable')
            end
        %1D finite difference approximation to update Tf
        for k = 1:100
            for j = 2:50
                Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
                T0(1,j) = Tij;
            end
        end

        qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
        Tf = T0(1,2);
end

t2 = 681:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);


Right = [TS1 TS2];
RightTime = [t1 t2];


Ts = 302;
Tf = 302;

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)


qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);


tiSi = 680;
T0 = Tf*ones(1,51);



for i = 1:TT %calculate temperature rise of the horizontal steel wall with
time
    if tiSi > 1360 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts+0.0003,'RelTol',1.6,'AbsTol',1.6); %how long
does it take for the steel to increase temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 0.0003K
```

```
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

    T0(1,1) = Ts;
    T0(1,51) = 302;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 680:1:1360;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));

Ts = TS1(1,681);
tiSi = 1360;

for i = 1:TT %calculate temperature drop of the horizontal steel wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003,'RelTol',1.6,'AbsTol',1.6); %how long
does it take for the steel to decrease temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1
```

```matlab
    Ts = Ts - 0.0003; %new steel temp

    d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
    Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
    Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
    ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for
exterior of steel box (W/m2K)

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t2 = 1361:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Top = [TS1 TS2];
TopTime = [t1 t2];

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

Ts = 302;
Tf = 302;
```

```matlab
tiSi = 1360;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the left steel wall with time
    if tiSi > 2040 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003,'RelTol',1.6,'AbsTol',1.6); %how long
does it take for the steel to increase temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 1360:1:2040;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));

Ts = TS1(1,681);
```

```matlab
tiSi = 2040;

for i = 1:TT %calculate temperature drop of the left steel wall with time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003,'RelTol',1.6,'AbsTol',1.6); %how long
does it take for the steel to decrease temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 302;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t2 = 2040:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Left = [TS1 TS2];
LeftTime = [t1 t2];

%interpolate the time and temperature arrays into 1x2261 arrays
day = 0:1:2260;
TR = interp1(RightTime,Right,day);
TF = interp1(TopTime,Top,day);
[LeftTime, index] = unique(LeftTime);
TL = interp1(LeftTime,Left(index),day);
```

```
figure
plot(day,TR,day,TF,day,TL)
```

Calculating T(x,y,t):

```
TT = 1;

W = 0.1976;    %scale steel box width (m)
H = 0.1976;    %scale steel box height (m)
L = 0.456;      %scalesteel box length (m)
wh = W/2;   %half the IBC width (m)
Ar = L*H;    %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;   %insolation (W/m2)
a = 0.05;   %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.95; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.55;   %thermal conductivity of feces (W/mK)
tC = 0.000635; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tibc = 300; %Ibc wall temperature (K)
Tf = 297; %Feces temperature (K)

pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)

mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)

cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)

A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;

S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/W;    %fecal conductance/half container thickness (W/m2K)

alpha = kf/(pF * cF); %Define thermal diffusivity for feces

T0 = Td1;
dx = W/200;

%2D finite difference approximation to calculate Tf(x,y,t)
for F = 1:TT
    for MM = 1:2261
```

```matlab
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end


        T0(:,201) = TR(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TL(1,MM);



        t1 = day(1,MM);
        dt = (t1 - t0)/200;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:200
            for i = 2:200
                for j = 2:200
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    end
end

az=0;
el=90;

T0(:,201) = TR(1,2261);
T0(1,:) = TF(1,2261);
T0(:,1) = TL(1,2261);

X = 0:0.1976/200:0.1976;
Y = 0.1976:-1*(0.1976/200):0;

figure
surf(X,Y,T0)
xlabel('x (m)','FontSize',18)
ylabel('y (m)','FontSize',18)
shading interp
colorbar
title(colorbar,'T(K)','FontSize',18)
view(az,el)
```

Un-insulated large-scale calculations:

Calculating T(t)

```
TT = 300000;

W = 2.6;    %steel box width (m)
dx = W/50;
H = 2.6;    %steel box height (m)
L = 6;      %steel box length (m)
wh = W/2;   %half the IBC width (m)
Ar = L*H;   %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
horizontal plate (m)
s = 1000;   %insolation (W/m2)
a = 0.1318; %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.87; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.5;  %thermal conductivity of feces (W/mK)
tC = 0.000635; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tf = 300; %Feces temperature (K)

pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)

mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)

cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)

A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;

S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/(2*dx);   %fecal conductance/half container thickness (W/m2K)

alpha = kf/(pF * cF); %Define thermal diffusivity for feces

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
```

```matlab
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

tiSi = 0;
T0 = Tf*ones(1,51);

for i = 1:TT %calculate temperature rise of the right steel wall with time
    if tiSi > 14400 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 0:1:14400;
TS1 = interp1(timeS,TempS,t1);
```

```matlab
timeS2 = zeros(1,1);
TempS2 = zeros(1,1);


qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));


Ts = TS1(1,14401);
tiSi = 14400;


for i = 1:TT %calculate temperature drop of the right steel wall with time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to decrease temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 minus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end


t2 = 14401:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Right = [TS1 TS2];
RightTime = [t1 t2];
```

```matlab
Ts = 300;
Tf = 300;

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

tiSi = 14400;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the top steel wall with time
    if tiSi > 28800 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
```

```matlab
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 14400:1:28800;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));

Ts = TS1(1,14401);
tiSi = 28800;

for i = 1:TT %calculate temperature drop of the top steel wall with time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to decrease temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 minus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
    Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
    Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
    ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for
exterior of steel box (W/m2K)

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end
```

```matlab
    qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end


t2 = 28801:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);


Top = [TS1 TS2];
TopTime = [t1 t2];


d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)


qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall


timeS = zeros(1,1);
TempS = zeros(1,1);


Ts = 300;
Tf = 300;


tiSi = 28800;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the left steel wall with time
    if tiSi > 43200 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;
```

```
    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 28800:1:43200;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));

Ts = TS1(1,14401);
tiSi = 43200;

for i = 1:TT %calculate temperature drop of the left steel wall with time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to decrease temp by 0.0003K
    tiSf = dt + tiSi; %time i
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 minus 0.0003K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
```

```matlab
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
    %1D finite difference approximation to update Tf
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar * (-1.*(b.*((Ts.^4) - Tao4)) - (ho.*(Ts-Tao)) - M*(Ts - 
Tf)));
    Tf = T0(1,2);
end


t2 = 43200:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);


Left = [TS1 TS2];
LeftTime = [t1 t2];


day = 0:1:86400;
TR = interp1(RightTime,Right,day);
TF = interp1(TopTime,Top,day);
[LeftTime, index] = unique(LeftTime);
TL = interp1(LeftTime,Left(index),day);


figure
plot(day,TR,day,TF,day,TL)
```

Calculating T(x,y,t) with no rotations:
```matlab
TT = 100;    %Number of simulated days


day = 0:1:86400;


W = 2.6;    %steel box width (m)
H = 2.6;    %steel box height (m)
L = 6;      %steel box length (m)
wh = W/2;  %half the IBC width (m)
Ar = L*H;   %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;  %insolation (W/m2)
a = 0.05;  %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.95; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
```

```matlab
kf = 0.6;   %thermal conductivity of feces (W/mK)
tC = 0.002; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tibc = 300; %Ibc wall temperature (K)
Tf = 300; %Feces temperature (K)

pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)

mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)

cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)

A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;

S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/W;    %fecal conductance/half container thickness (W/m2K)

alpha = kf/(pF * cF); %Define thermal diffusivity for feces

T0 = 300 * ones(11);

 for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        T0(:,11) = TR(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TL(1,MM);

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
```

```matlab
                        Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                        if isnan(Tij) > 0
                            break
                        end
                        T0(i,j) = Tij;
                    end
                end
            end
            if F == TT
                if MM == 21600
                    M6 = T0;
                  elseif MM == 43200
                    M12 = T0;
                  elseif MM == 64800
                    M18 = T0;
            end
            end
        end
    end
end


az=0;
el=90;


X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;


figure
surf(X,Y,T0)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(TT),' days'])
view(az,el)
```

Calculating T(x,y,t) with rotations:

```matlab
TT = 100; %Number of simulated days before polar rotation


day = 0:1:86400;


W = 2.6;    %steel box width (m)
H = 2.6;    %steel box height (m)
L = 6;      %steel box length (m)
wh = W/2;   %half the IBC width (m)
Ar = L*H;    %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;   %insolation (W/m2)
a = 0.05;   %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.95; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
```

```matlab
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.6;  %thermal conductivity of feces (W/mK)
tC = 0.002; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tibc = 300; %Ibc wall temperature (K)
Tf = 300; %Feces temperature (K)

pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)

mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)

cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)

A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;

S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/W;   %fecal conductance/half container thickness (W/m2K)

alpha = kf/(pF * cF); %Define thermal diffusivity for feces

T0 = 300 * ones(11);

 for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        T0(:,11) = TR(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TL(1,MM);

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end
```

```matlab
        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
        if F == TT
            if MM == 21600
                M6 = T0;
              elseif MM == 43200
                M12 = T0;
              elseif MM == 64800
                M18 = T0;
          end
        end
    end
end

az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

figure
surf(X,Y,T0)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(TT),' days'])
view(az,el)

for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        T0(:,11) = TL(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TR(1,MM);

        t1 = day(1,MM);
        dt = (t1 - t0)/10;
```

```
        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j)))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    if F == TT
        if MM == 21600
            M6 = T0;
        elseif MM == 43200
            M12 = T0;
        elseif MM == 64800
            M18 = T0;
        end
    end
   end
end


az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

figure
surf(X,Y,T0)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(2*TT),'
days'])
view(az,el)
```

Calculating T(x,y,t) under 1st order insulation conditions:

```
TT = 80; %number of days before a polar rotation

day = 0:1:86400;
```

```matlab
W = 2.6;    %steel box width (m)
H = 2.6;    %steel box height (m)
L = 6;      %steel box length (m)
wh = W/2;   %half the IBC width (m)
Ar = L*H;   %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;   %insolation (W/m2)
a = 0.05;   %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.95; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.6;  %thermal conductivity of feces (W/mK)
tC = 0.002; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tibc = 300; %Ibc wall temperature (K)
Tf = 300; %Feces temperature (K)


pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)


mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)


cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)


A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;


S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/W;   %fecal conductance/half container thickness (W/m2K)


alpha = kf/(pF * cF); %Define thermal diffusivity for feces


T0 = 300 * ones(11);


for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end
```

```matlab
        T0(:,11) = TR(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TL(1,MM);
        T0(11,:) = T0(10,:);

        if MM > 43200
            T0(:,11) = T0(:,10);
            T0(1,:) = T0(2,:);
            T0(:,1) = T0(:,2);
            T0(11,:) = T0(10,:);
        end

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
        if F == TT
            if MM == 21600
                M6 = T0;
              elseif MM == 43200
                M12 = T0;
              elseif MM == 64800
                M18 = T0;
          end
        end
    end
end

az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

figure
surf(X,Y,T0)
xlabel('x (m)')
```

```matlab
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(TT),' days'])
view(az,el)

for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        T0(:,11) = TL(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TR(1,MM);
        T0(11,:) = T0(10,:);

        if MM > 43200
            T0(:,11) = T0(:,10);
            T0(1,:) = T0(2,:);
            T0(:,1) = T0(:,2);
            T0(11,:) = T0(10,:);
        end

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    if F == TT
        if MM == 21600
            M6 = T0;
        elseif MM == 43200
            M12 = T0;
        elseif MM == 64800
```

```matlab
            M18 = T0;
          end
      end
    end
end


az=0;
el=90;


X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;


figure
surf(X,Y,T0)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(2*TT),'
days'])
view(az,el)
```

Calculating T(x,y,t) under 2nd order insulation conditions with 3 polar rotations:

```matlab
TT = 40; %Number of days per each of the four orientations


day = 0:1:86400;


W = 2.6;   %steel box width (m)
H = 2.6;   %steel box height (m)
L = 6;     %steel box length (m)
wh = W/2;  %half the IBC width (m)
Ar = L*H;   %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;  %insolation (W/m2)
a = 0.05;  %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.95; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.6;  %thermal conductivity of feces (W/mK)
tC = 0.002; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tibc = 300; %Ibc wall temperature (K)
Tf = 300; %Feces temperature (K)


pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)
```

```matlab
mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)


cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)


A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;


S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/W;    %fecal conductance/half container thickness (W/m2K)


alpha = kf/(pF * cF); %Define thermal diffusivity for feces


T0 = 300 * ones(13);


for F = 1:TT
   for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        if MM < 43200
            T0(:,12) = TR(1,MM);
            T0(1,:) = TF(1,MM);
            T0(:,1) = TL(1,MM);
            T0(:,13) = TR(1,MM);
            T0(2,:) = TF(1,MM);
            T0(:,2) = TL(1,MM);
            T0(13,:) = T0(11,:);
        end
        if MM > 43200
            T0(:,13) = T0(:,11);
            T0(1,:) = T0(3,:);
            T0(:,1) = T0(:,3);
            T0(13,:) = T0(11,:);
        end

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end
    if MM < 43200
```

```
        for k = 1:10
            for i = 3:11
                for j = 3:11
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    else
        for k = 1:10
            for i = 2:12
                for j = 2:12
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    end
  end
end

az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

T0cut1 = T0(2:12,2:12);

figure
surf(X,Y,T0cut1)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(TT),' days'])
view(az,el)

for F = 1:TT
   for MM = 1:86401
       if MM == 1
           t0 = 0;
       else
           t0 = day(1,MM-1);
       end
       if MM < 43200
           T0(:,12) = TL(1,MM);
```

```
            T0(1,:)  = TF(1,MM);
            T0(:,1)  = TR(1,MM);
            T0(:,13) = TL(1,MM);
            T0(2,:)  = TF(1,MM);
            T0(:,2)  = TR(1,MM);
            T0(13,:) = T0(11,:);
        end
        if MM > 43200
            T0(:,13) = T0(:,11);
            T0(1,:)  = T0(3,:);
            T0(:,1)  = T0(:,3);
            T0(13,:) = T0(11,:);
        end

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

    if MM < 43200
        for k = 1:10
            for i = 3:11
                for j = 3:11
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    else
        for k = 1:10
            for i = 2:12
                for j = 2:12
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    end
  end
end

az=0;
el=90;
```

```matlab
X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

T0cut2 = T0(2:12,2:12);

figure
surf(X,Y,T0cut2)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(2*TT),'
days'])
view(az,el)

for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        if MM < 43200
            T0(:,12) = TR(1,MM);
            T0(1,:) = TF(1,MM);
            T0(:,1) = TL(1,MM);
            T0(:,13) = TR(1,MM);
            T0(2,:) = TF(1,MM);
            T0(:,2) = TL(1,MM);
            T0(13,:) = T0(11,:);
        end
        if MM > 43200
            T0(:,13) = T0(:,11);
            T0(1,:) = T0(3,:);
            T0(:,1) = T0(:,3);
            T0(13,:) = T0(11,:);
        end

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end
    if MM < 43200
        for k = 1:10
            for i = 3:11
                for j = 3:11
```

```
                         Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                            if isnan(Tij) > 0
                                break
                            end
                            T0(i,j) = Tij;
                    end
                end
            end
    else
        for k = 1:10
            for i = 2:12
                for j = 2:12
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                        if isnan(Tij) > 0
                            break
                        end
                        T0(i,j) = Tij;
                end
            end
        end
    end
    if MM == 32000
      TopTemp1 = T0;
    end
    end
end

az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

T0cut3 = T0(2:12,2:12);

figure
surf(X,Y,T0cut3)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(3*TT),'
days'])
view(az,el)

for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end
        if MM < 43200
```

```
            T0(:,12) = TL(1,MM);
            T0(1,:) = TF(1,MM);
            T0(:,1) = TR(1,MM);
            T0(:,13) = TL(1,MM);
            T0(2,:) = TF(1,MM);
            T0(:,2) = TR(1,MM);
            T0(13,:) = T0(11,:);
        end
        if MM > 43200
             T0(:,13) = T0(:,11);
             T0(1,:) = T0(3,:);
             T0(:,1) = T0(:,3);
             T0(13,:) = T0(11,:);
        end

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

    if MM < 43200
        for k = 1:10
            for i = 3:11
                for j = 3:11
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    else
        for k = 1:10
            for i = 2:12
                for j = 2:12
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    end
    if MM == 32000
      TopTemp = T0;
    end
  end
```

```matlab
end

az=0;
el=90;


X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;


T0cut4 = T0(2:12,2:12);

figure
surf(X,Y,T0cut4)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(4*TT),'
days'])
view(az,el)

TopTemp = TopTemp(2:12,2:12);
TopTemp1 = TopTemp1(2:12,2:12);

figure
surf(X,Y,TopTemp)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
view(az,el)

disp(T0cut1)
disp(T0cut2)
disp(T0cut3)
disp(T0cut4)
```

Calculating T(t) utilizing R-value insulation

```matlab
TT = 300000;

W = 2.6;    %steel box width (m)
dx = W/50;
H = 2.6;    %steel box height (m)
L = 6;      %steel box length (m)
wh = W/2;   %half the IBC width (m)
Ar = L*H;    %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;   %insolation (W/m2)
a = 0.1318;  %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.87; %steel emissivity (-)
```

```
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
vf = 350/1300;
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.5;  %thermal conductivity of feces (W/mK)
tC = 0.000635; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tf = 300; %Feces temperature (K)
RR = 1/0.85; %R-value


pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)


mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)


cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)


A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;


S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
Tf4 = (Tf)^4;
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/(2*dx);   %fecal conductance/half container thickness (W/m2K)



alpha = kf/(pF * cF); %Define thermal diffusivity for feces
Prf = vf/alpha;


d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)


qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall


timeS = zeros(1,1);
TempS = zeros(1,1);


tiSi = 0;
T0 = Tf*ones(1,51);
```

```matlab
for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 14400 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 0:1:14400;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 14400;
```

```matlab
for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 86400
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 14400:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Right = [TS1 TS2];
RightTime = [t1 t2];

Ts = 300;
Tf = 300;

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
```

```matlab
TempS = zeros(1,1);

tiSi = 14400;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 28800 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 14400:1:28800;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);
```

```matlab
qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 28800;

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 100800
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 28800:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Top = [TS1 TS2];
TopTime = [t1 t2];

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)
```

```matlab
qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

Ts = 300;
Tf = 300;

tiSi = 28800;
T0 = Tf*ones(1,51);

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 43200 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end
```

```matlab
t1 = 28800:1:43200;
TS1 = interp1(timeS,TempS,t1);


timeS2 = zeros(1,1);
TempS2 = zeros(1,1);


qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));


Ts = TS1(1,14401);
tiSi = 43200;


for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 115200
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 43200:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);


Left = [TS1 TS2];
LeftTime = [t1 t2];


dayR = 0:1:86400;
dayF = 0:1:100800;
```

```matlab
dayL = 0:1:115200;
[RightTime, index] = unique(RightTime);
TR = interp1(RightTime,Right(index),dayR);
[TopTime, index] = unique(TopTime);
TF = interp1(TopTime,Top(index),dayF);
[LeftTime, index] = unique(LeftTime);
TL = interp1(LeftTime,Left(index),dayL);


Ts = 306; %Steel temperature (K)
Tf = 306; %Feces temperature (K)

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

tiSi = 86400;
T0 = Tf*ones(1,51);

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 100800 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
```

```matlab
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 86400:1:100800;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 100800;

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 || tiSi > 172800
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
```

```matlab
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 100800:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Right = [TS1 TS2];
RightTime = [t1 t2];

Ts = 306;
Tf = 306;

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

tiSi = 100800;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 115200 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)
```

```matlab
    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end


    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end


t1 = 100800:1:115200;
TS1 = interp1(timeS,TempS,t1);


timeS2 = zeros(1,1);
TempS2 = zeros(1,1);


qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));


Ts = TS1(1,14401);
tiSi = 115200;


for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 187200
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end
```

97

```
    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 115200:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);


Top = [TS1 TS2];
TopTime = [t1 t2];


d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall


timeS = zeros(1,1);
TempS = zeros(1,1);


Ts = 306;
Tf = 306;


tiSi = 115200;
T0 = Tf*ones(1,51);


for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 129600 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1
```

```matlab
    Ts = Ts + 0.0003; %new steel temp


    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;


    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end


    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end


t1 = 115200:1:129600;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 129600;

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 201600
        break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp
```

```matlab
    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end


    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end


    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 129600:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Left = [TS1 TS2];
LeftTime = [t1 t2];

day2R = 86400:1:172800;
day2F = 86400:1:187200;
day2L = 86400:1:201600;
[RightTime, index] = unique(RightTime);
TR2 = interp1(RightTime,Right(index),day2R);
[TopTime, index] = unique(TopTime);
TF2 = interp1(TopTime,Top(index),day2F);
[LeftTime, index] = unique(LeftTime);
TL2 = interp1(LeftTime,Left(index),day2L);

Ts = 307; %Steel temperature (K)
Tf = 307; %Feces temperature (K)

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);
```

```matlab
tiSi = 172800;
T0 = Tf*ones(1,51);

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 187200 || isnan(Ts) > 0 || isnan(tiSi) > 0
        break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 172800:1:187200;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 187200;
```

```matlab
for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 || tiSi > 259200
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 187200:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Right = [TS1 TS2];
RightTime = [t1 t2];

Ts = 307;
Tf = 307;

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall
```

```matlab
timeS = zeros(1,1);
TempS = zeros(1,1);

tiSi = 187200;
T0 = Tf*ones(1,51);



for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 201600 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

d = (g*(1/(Tao))*((Lc)^3)*((Pr)^2))/(v^2); %Rayliegh per temperature
difference (1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
Nuo = 0.15*((Ro)^(0.33)); %Nusselt outside
ho = ((Nuo)*(kair))/(Lc); %Convective heat transfer coefficient for exterior
of steel box (W/m2K)

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end

t1 = 187200:1:201600;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
```

```
TempS2 = zeros(1,1);

qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 201600;

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 259200
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 201600:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Top = [TS1 TS2];
TopTime = [t1 t2];

d = (g*(1/(Tao))*(H^3)*((Pr)))/(v^2); %Rayliegh per temperature difference
(1/K)
Ro = d*abs(Ts-Tao); %Rayliegh number exterior (-)
j2 = (0.387)*((Ro)^(0.167)); %Empirical turbulent Nusselt numerator
l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296); %Empirical turbulent Nusselt
denominator
Nuo = (0.825 + ((j2)/(l2)))^2; %Nusselt outside
```

```matlab
ho = ((Nuo)*(kair))/H; %Convective heat transfer coefficient for exterior of
steel box (W/m2K)

qS = @(Ts) 1./(Ar .* (S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts - Tao)) - M.*(Ts -
Tf))); %net heat transfer per unit area to the steel wall

timeS = zeros(1,1);
TempS = zeros(1,1);

Ts = 307;
Tf = 307;

tiSi = 201600;
T0 = Tf*ones(1,51);

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if tiSi > 216000 || isnan(Ts) > 0 || isnan(tiSi) > 0
       break
    end

    dt = A * integral(qS,Ts,Ts+0.0003); %how long does it take for the steel
to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS(1,i) = tiSf; %declare time i
    TempS(1,i) = Ts + 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts + 0.0003; %new steel temp

    Ro = d * (abs(Ts-Tao));
    j2 = (0.387)*((Ro)^(0.167));
    l2 = (1+((0.492/(Pr))^(0.5625)))^(0.296);
    Nuo = (0.825 + ((j2)/(l2)))^2;
    ho = ((Nuo)*(kair))/H;

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) 1./(Ar*(S - (b.*((Ts.^4)-Tao4)) - (ho.*(Ts-Tao)) - M*(Ts -
Tf)));
    Tf = T0(1,2);
end
```

```
t1 = 201600:1:216000;
TS1 = interp1(timeS,TempS,t1);

timeS2 = zeros(1,1);
TempS2 = zeros(1,1);

qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));

Ts = TS1(1,14401);
tiSi = 216000;

for i = 1:TT %calculate temperature rise of the steel, air, and IBC wall with
time
    if Ts < 297 || isnan(Ts) > 0 || isnan(tiSi) > 0 ||tiSi > 259200
       break
    end

    dt = A * integral(qS,Ts,Ts - 0.0003); %how long does it take for the
steel to increase temp by 2K
    tiSf = dt + tiSi; %time i minus time i-1
    timeS2(1,i) = tiSf; %declare time i
    TempS2(1,i) = Ts - 0.0003; %temp i-1 plus 2K
    tiSi = tiSf; %set time i as new time i-1

    Ts = Ts - 0.0003; %new steel temp

    T0(1,1) = Ts;
    T0(1,51) = 300;
    dt = dt/100;
    Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.5
            disp('unstable')
        end

    for k = 1:100
        for j = 2:50
            Tij = T0(1,j) + Z * (T0(1,j+1) + T0(1,j-1) - 2*T0(1,j));
            T0(1,j) = Tij;
        end
    end

    qS = @(Ts) -1./(Ar * (M*(Ts - Tf) + RR*(Ts - Tao)));
    Tf = T0(1,2);
end

t2 = 216000:1:tiSf;
[timeS2, index] = unique(timeS2);
TS2 = interp1(timeS2,TempS2(index),t2);

Left = [TS1 TS2];
LeftTime = [t1 t2];

day3 = 172800:1:259200;
```

```
[RightTime, index] = unique(RightTime);
TR3 = interp1(RightTime,Right(index),day3);
[TopTime, index] = unique(TopTime);
TF3 = interp1(TopTime,Top(index),day3);
[LeftTime, index] = unique(LeftTime);
TL3 = interp1(LeftTime,Left(index),day3);


plot(dayR,TR,dayF,TF,dayL,TL,day2R,TR2,day2F,TF2,day2L,TL2,day3,TR3,day3,TF3,
day3,TL3)
```

Calculating T(x,y,t) for R-value insulation model

```
TT = 40; %Number of days in the simulation before a polar rotation


day = 0:1:86400;


W = 2.6;    %steel box width (m)
H = 2.6;    %steel box height (m)
L = 6;      %steel box length (m)
wh = W/2;   %half the IBC width (m)
Ar = L*H;    %large box surface area (m)
Lc = Ar/(2*W + 2*L); %Characteristic length for natural convection of
downward facing heated plate (m)
s = 1000;   %insolation (W/m2)
a = 0.05;   %steel albedo (-)
SB = 5.67*(10^-8); %Stefan Boltzmann constant (W/m2K4)
e = 0.95; %steel emissivity (-)
Tao = 300; %air temperature outside (K)
Pr = 0.71; %Prandtl Number for air
g = 9.81; %acceleration due to gravity (m/s2)
v = 0.000015; %kinematic viscosity (m2/s)
kair = 0.028; %thermal conductivity of air (W/mK)
ks = 40; %thermal conductivity of steel (W/mK)
kf = 0.6;  %thermal conductivity of feces (W/mK)
tC = 0.002; %steel box thickness (m)
Ts = 300; %Steel temperature (K)
Tibc = 300; %Ibc wall temperature (K)
Tf = 300; %Feces temperature (K)


pS = 7700; %steel density (kg/m3)
pF = 1300; %feces density (kg/m3)


mS = pS*tC*Ar; %steel wall mass (kg)
mF = pF*W*Ar; %feces half mass (kg)


cS = 470; %steel specific heat (J/kgK)
cF = 2700; %feces specific heat (J/kgK)


A = (mS*cS); %mass of steel * specific heat per area  (J/K)
D = (mF*cF); %mass of feces * specific heat per area (J/K)
DR = 1./D;


S = s*(1-a); %Shortwave absorbed (W/m2)
b = SB * e; %Stef Boltz times emissivity of steel (W/m2K4)
```

```
Tao4 = (Tao)^4; %exterior air temperature to the fourth (K4)
K = (ks/tC); %steel conductance/wall thickness (W/m2K)
M = kf/W;    %fecal conductance/half container thickness (W/m2K)

alpha = kf/(pF * cF); %Define thermal diffusivity for feces

T0 = 300 * ones(11);

    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        T0(:,11) = TR(1,MM);
        T0(1,:) = TF(1,MM);
        T0(:,1) = TL(1,MM);
        T0(11,:) = T0(10,:);

        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j)))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    end
 for F = 1:TT
    for MM = 1:86401
        if MM == 1
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end

        T0(:,11) = TR2(1,MM);
        T0(1,:) = TF2(1,MM);
        T0(:,1) = TL2(1,MM);
        T0(11,:) = T0(10,:);
```

```matlab
        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
        if F == TT
            if MM == 21600
                M6 = T0;
              elseif MM == 43200
                M12 = T0;
              elseif MM == 64800
                M18 = T0;
          end
        end
    end
end

az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

figure
surf(X,Y,T0)
xlabel('x (m)')
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(1+TT),'
days'])
view(az,el)

for F = 1:TT
    for MM = 1:86401
        if MM == 1
```

```matlab
            t0 = 0;
        else
            t0 = day(1,MM-1);
        end


        T0(:,11) = TL2(1,MM);
        T0(1,:) = TF2(1,MM);
        T0(:,1) = TR2(1,MM);
        T0(11,:) = T0(10,:);



        t1 = day(1,MM);
        dt = (t1 - t0)/10;

        dx = W/10;

        Z = ((alpha) * (dt))/(dx ^ 2);
        if Z > 0.25
            print 'unstable'
        end

        for k = 1:10
            for i = 2:10
                for j = 2:10
                    Tij = T0(i,j) + Z * ((T0(i-1,j)+T0(i+1,j)+T0(i,j-
1)+T0(i,j+1)-(4*T0(i,j))));
                    if isnan(Tij) > 0
                        break
                    end
                    T0(i,j) = Tij;
                end
            end
        end
    if F == TT
        if MM == 21600
            M6 = T0;
        elseif MM == 43200
            M12 = T0;
        elseif MM == 64800
            M18 = T0;
        end
    end
   end
end


az=0;
el=90;

X = 0:2.6/10:2.6;
Y = 2.6:-1*(2.6/10):0;

figure
surf(X,Y,T0)
xlabel('x (m)')
```

```
ylabel('y (m)')
shading interp
colorbar
title(colorbar,'T(K)')
title(['Temperature Distribution within the Bag after ',num2str(1 + 2*TT),'
days'])
view(az,el)
```

Appendix 3: Temperature Sensor Coding and Circuitry

The temperature distribution throughout the fecal simulant was measured during the laboratory experiments using DS18b20 digital temperature sensors. The measurements were collected using an Arduino UNO and were recorded by a computer using the program, CoolTermWin. The code used to communicate with the sensors is presented below along with the circuitry. The codes were modified from code found at: https://randomnerdtutorials.com/guide-for-ds18b20-temperature-sensor-with-arduino/. These codes were modified and sent to the Arduino UNO using the Arduino IDE.

1: The following code was used to learn the individual addresses of each sensor that was used in the experiment.

```
#include <OneWire.h>

// Based on the OneWire library example

OneWire ds(2);  //data wire connected to GPIO15

void setup(void) {
  Serial.begin(9600);
}

void loop(void) {
  byte i;
  byte addr[8];

  if (!ds.search(addr)) {
    Serial.println(" No more addresses.");
    Serial.println();
    ds.reset_search();
    delay(250);
    return;
  }
  Serial.print(" ROM =");
  for (i = 0; i < 8; i++) {
    Serial.write(' ');
    Serial.print(addr[i], HEX);
  }
}
```

2: The following code was used to retrieve the measurements from all the sensors used. 16 sensors were wired in the same way as the schematic shown below (image source: https://randomnerdtutorials.com/guide-for-ds18b20-temperature-sensor-with-arduino/). The addresses in the code below are specific to the sensors used for this analysis. Utilizing other DS18b20 sensors would require reacquisition of the addresses. The pull-up resistor in the schematic is 4.7 kΩ. The digital pin of the Arduino used in the experiment was pin 2.
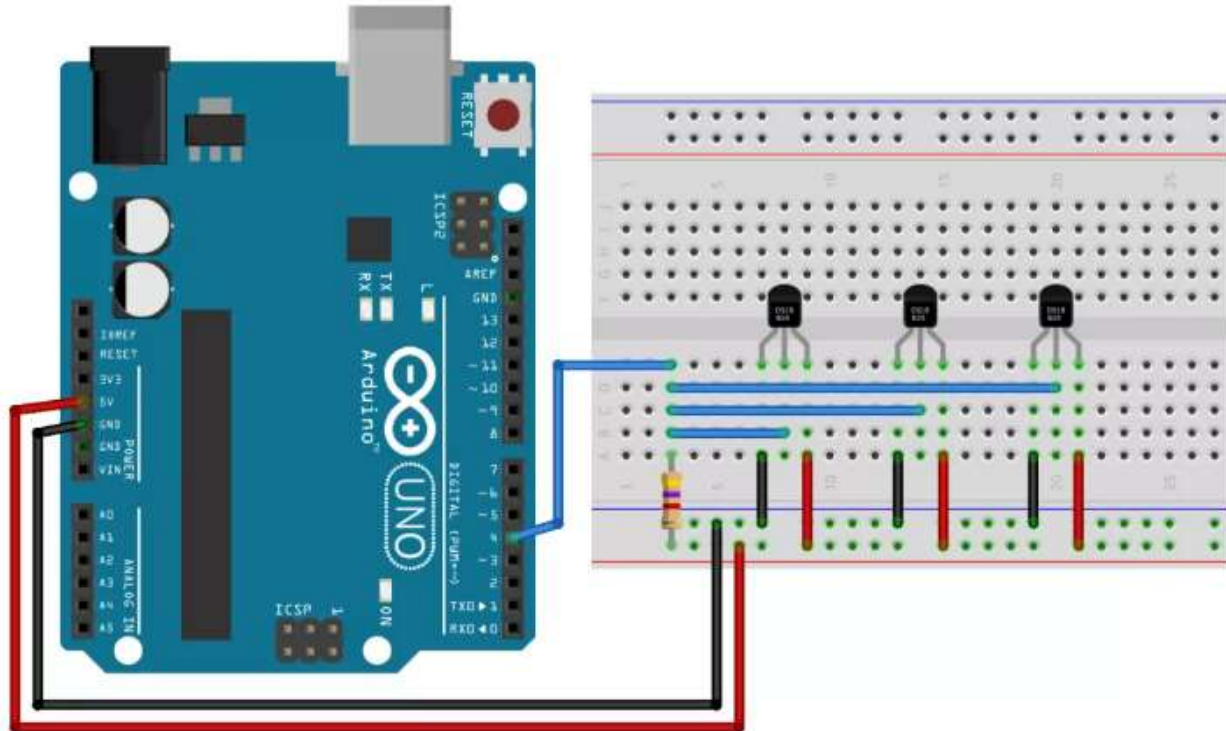
```
#include <OneWire.h>
#include <DallasTemperature.h>

// Data wire is connected to GPIO15
#define ONE_WIRE_BUS 2
// Setup a oneWire instance to communicate with a OneWire device
OneWire oneWire(ONE_WIRE_BUS);
// Pass our oneWire reference to Dallas Temperature sensor
DallasTemperature sensors(&oneWire);

uint8_t sensor1[8] = { 0x28, 0x7C, 0x27, 0x79, 0x97, 0x1, 0x3, 0x1D };
uint8_t sensor2[8] = { 0x28, 0x77, 0x93, 0x79, 0x97, 0x6, 0x3, 0xDB };
uint8_t sensor3[8] = { 0x28, 0x71, 0x93, 0x79, 0x97, 0x6, 0x3, 0x69 };
uint8_t sensor4[8] = { 0x28, 0x2F, 0x8A, 0x79, 0x97, 0x6, 0x3, 0xDB };
uint8_t sensor5[8] = { 0x28, 0x2B, 0x14, 0x79, 0x97, 0x7, 0x3, 0xD6 };
uint8_t sensor6[8] = { 0x28, 0x31, 0x95, 0x79, 0x97, 0x7, 0x3, 0x44 };
uint8_t sensor7[8] = { 0x28, 0x46, 0x39, 0x79, 0x97, 0x7, 0x3, 0x4B };
uint8_t sensor8[8] = { 0x28, 0xF7, 0xAE, 0x79, 0x97, 0x6, 0x3, 0x59 };
uint8_t sensor9[8] = { 0x28, 0xC8, 0xC7, 0x79, 0x97, 0x6, 0x3, 0x72 };
uint8_t sensor10[8] = { 0x28, 0x5, 0x88, 0x79, 0x97, 0x7, 0x3, 0xE5 };
uint8_t sensor11[8] = { 0x28, 0x1B, 0x9F, 0x79, 0x97, 0x6, 0x3, 0x44 };
uint8_t sensor12[8] = { 0x28, 0xD0, 0xEC, 0x79, 0x97, 0x6, 0x3, 0x0 };
uint8_t sensor13[8] = { 0x28, 0x7D, 0xE, 0x79, 0x97, 0x1, 0x3, 0x21 };
uint8_t sensor14[8] = { 0x28, 0x40, 0xC7, 0x79, 0x97, 0x6, 0x3, 0x39 };
uint8_t sensor15[8] = { 0x28, 0x48, 0xA1, 0x79, 0x97, 0x6, 0x3, 0x15 };
uint8_t sensor16[8] = { 0x28, 0x73, 0x5A, 0x79, 0x97, 0x2, 0x3, 0xED };
uint8_t sensor18[8] = {0x28, 0xF3, 0x27, 0x79, 0x97, 0x1, 0x3, 0xD3 };
void setup(void){
  Serial.begin(9600);
  sensors.begin();
}
```

```
void loop(void){
  sensors.requestTemperatures(); // Send the command to get temperatures

  Serial.print(",1, ");
  Serial.println(sensors.getTempC(sensor1));

  Serial.print(",2, ");
  Serial.println(sensors.getTempC(sensor2));

  Serial.print(",3, ");
  Serial.println(sensors.getTempC(sensor3));

  Serial.print(",4, ");
  Serial.println(sensors.getTempC(sensor4));

  Serial.print(",5, ");
  Serial.println(sensors.getTempC(sensor5));

  Serial.print(",6, ");
  Serial.println(sensors.getTempC(sensor6));

  Serial.print(",7, ");
  Serial.println(sensors.getTempC(sensor7));

  Serial.print(",8, ");
  Serial.println(sensors.getTempC(sensor8));

  Serial.print(",9, ");
  Serial.println(sensors.getTempC(sensor9));

  Serial.print(",10, ");
  Serial.println(sensors.getTempC(sensor10));

  Serial.print(",11, ");
  Serial.println(sensors.getTempC(sensor11));

  Serial.print(",12, ");
  Serial.println(sensors.getTempC(sensor12));

  Serial.print(",13, ");
  Serial.println(sensors.getTempC(sensor13));

  Serial.print(",14, ");
  Serial.println(sensors.getTempC(sensor14));

  Serial.print(",15, ");
  Serial.println(sensors.getTempC(sensor15));

  Serial.print(",16, ");
  Serial.println(sensors.getTempC(sensor16));

  Serial.print(",18, ");
  Serial.println(sensors.getTempC(sensor18));

  delay(60000);
}
```

Chapter 8: References

Anderson Jr., J. D. (1995). *Computational Fluid Dynamics: The Basics with Applications*. New

    York: McGraw-Hill Inc.

Colón, J., Forbis-Stokes, A. A., & Deshusses, M.A. (2015). Anaerobic digestion   of

    undiluted simulant human excreta for sanitation and energy recovery in less-developed

    countries. *Energy for Sustainable Development, 29,* 57-64.

Dai, W. (2010). A new accurate finite difference scheme for Neumann (insulated)

    boundary condition of heat conduction. *International Journal of Thermal Sciences, 49,*

    571-579.

Davis, J. & Lambert, R. (2002). *Engineering in Emergencies: A Practical Guide for*

    Relief Workers. Rugby: Practical Action Publishing.

Feachem, R.G., Bradley, D.J., Garelick, H., & Mara, D. D. (1983).

    Saniation and Disease: Health Aspects of Excreta and Wastewater Management.

    Washington, DC: THE WORLD BANK.

Foote, A. M., Woods, E., Fredes, F., Leon, J. S. (2017). Rendering fecal waste safe for reuse

    via a cost-effective solar concentrator. *Journal of Water, Sanitation and Hygiene for*

    Development, 07.2, 252-259.

Guzha, E., Nhapi, I., & Rockstrom, J. (2005). An assessment of the effect of human faeces

    and urine on maize production and water productivity. *Physics and Chemistry of   the*

    Earth, *30,* 840-845.

Hagen, K. D. (1999). *Heat Transfer with Applications.* Upper Saddle River: Prentice Hall

    Inc.

Iwabuchi, K., Kimura, T., & Otten, L. (1999). Effect of volumetric water content on

    thermal properties of dairy cattle feces mixed with sawdust. *Bioresource Technology,*

    70.3, 293-297.

Jenkins, J. (2005). *The Humanure Handbook: A Guide to Composting Human Manure*.

    White River Junction: Chelsea Green Publishing.

Langergraber, G. & Muellegger, E. (2005). Ecological Sanitation—a way to solve

    global sanitation problems? *Environment International, 31,* 433-444.

Martin, E. (2016, June 14). This sanitation startup in Kenya turns poop into a sustainable source

    of fuel. *Business Insider.* Retrieved from https://www.businessinsider.com.

Mihelcic, J. R., Fry, L. M., Myre, E. A., Phillips, L. D., & Barkdoll,

    Brian D. (2009). *Field Guide to Environmental Engineering for Development Workers:*

    Water, Sanitation, and Indoor Air. Reston: American Society of Civil Engineers.

Murphy, C. M. (2015) Solids Accumulation Rates of Latrines at Rural Schools in Nimba

    County, Liberia. *Graduate Theses and Dissertations.* Retrieved from

    https://scholarcommons.usf.edu/etd/5545.

Naughton, C. C., Akers, P., Yoder, D., Baer, R., & Mihelcic, J.R. (2018). Can Sanitation

    Technology Play a Role in User Perceptions of Resource Recovery? An Evaluation of

    Composting Latrine Use in Developing World Communities in Panama. *Environmental*

    Science & Technology, 52, 11803-11812.

Nave, R. (2000). *Radiative Cooling Time.* Retrieved from

    http://hyperphysics.phy-astr.gsu.edu/hbase/thermo/cootime.html

Niwagaba, C., Nalubega, M., Vinnerås, B., Sundberg, C., Jönsson, H. (2003). Bench-scale composting of source-separated human faeces for sanitation. *Waste Management, 29,* 585-589.

Orner, K. D., & Mihelcic, J. R. (2018). A review of sanitation technologies to achieve multiple sustainable development goals that promote resource recovery. *Environmental Science Water Research & Technology*, *4,* 16 -32.

Ostace, G.S., Biegler, L.T., Grossman, I.E., Stoltz, C., & Weinstein, B. (2013). Heat transfer model of large shipping containers. *Carnegie Mellon University*. Retrieved from http://egon.cheme.cmu.edu/ewo/docs/PG_%20Biegler_3_2013.pdf.

Pollution Research Group, University of KwaZulu-Natal (2014). *Selection of synthetic sludge simulant for the Bill and Melinda Gates Foundation's Reinvent the Toilet Fair: India 2014.*

Progress on drinking water, sanitation and hygiene: 2017 update and SDG baselines. Geneva: World Health Organization (WHO) and the United Nations Children's Fund (UNICEF), 2017. Licence: CC BY-NC-SA 3.0 IGO.

Rennie, T. J., Baldé, H., Gordon, R. J., Smith, W. N., & VanderZaag, A. C. (2017). A 3-D model to predict the temperature of liquid manure within storage tanks. *Biosystems Engineering, 163*, 50-65.

Rose, C., Parker, A., Jefferson, B., & Cartmell, E. (2015). The Characterization of Feces and Urine: A Review of the Literature to Inform Advanced Treatment Technology. *Critical Reviews in Environmental Science and Technology, 45,* 1827-1879.

Ruiz-Espinoza, J. E., Méndez-Contreras, J.M., Alvarado-Lassman, A., & Martínez-Delgadillo, S.A. (2012). Effect of low temperature thermal pre-treatment on the solubilization of organic matter, pathogen inactivation and mesophilic anaerobic digestion of poultry sludge. *Journal of Environmental Science and Health, Part A 47,* 1795-1802.

Sauer, T. D. (2006). *Numerical Analysis.* Boston: Pearson Addison Wesley.

Schroeder, D. V. (2000). *An Introduction to Thermal Physics.* San Francisco: Addison Wesley Longman.

Simha, P., Lalander, C., Ramanthan, A., Vijayalakshmi, C., McConville, J. R., Vinnerås, B., & Ganesapillai, M. (2018). What do consumers think about recycling human urine as fertiliser? Perceptions and attitudes of a university community in South India. *Water Research, 143,* 527-538.

Skoglund, V. J. (1967). *Similitude: Theory and Applications.* Scranton: International Textbook Company.

Sustainable Development Goals: Knowledge Platform. (n.d.). Retrieved January 15, 2019 from https://sustainabledevelopment.un.org/?menu=1300.

Tan, Q. (2011). *Dimensional Analysis with Case Studies in Mechanics.* Heidelberg: Springer.

Tumwebaze, I. K., Orach, C. G., Nakayaga, J. K., Karamagi, C., Luethi, C., & Niwagaba, C. (2011). Ecological sanitation coverage and factors affecting its uptake in Kabale municipality, western Uganda. *International Journal of Environmental Health Research, 21.4,* 294-305.

Vinnerås, B., Björklund, A., Jönsson, H. (2003). Thermal composting of faecal matter as

treatment and possible disinfection method—laboratory-scale and pilot-scale studies.

Bioresource Technology, 88, 47-54.

You, D., Jones, G., & Wardlaw, T. (2011). *Levels & Trends in Child*

Mortality: Report 2011. New York: UNICEF.

Chapter 9: Vita

Ryan Homeyer
(315) 727-1500 | rjhomeye@syr.edu

Education

Syracuse University, College of Engineering and Computer Science, Syracuse, NY
M.S. Environmental Engineering, August 2019
- GPA: 3.900
- Research Experience in Thermodynamic Modelling, Humanitarian Engineering, and Environmental Sensor Design
- Courses in hydrology, fluid dynamics, engineering design, and environmental chemistry

Le Moyne College, College of Arts and Sciences, Syracuse, NY
B.S. Physics, May 2017
Minor: Mathematics
- GPA: 3.893 · Graduated Summa Cum Laude · Recognized on Dean's List for each semester attended
- Awarded the Physics Program Medal for Outstanding Performance in Physics
- Courses in classical, quantum, statistical, and computational physics and electrodynamics

Research/Lab Experience

Syracuse University, Department of Civil and Environmental Engineering, Syracuse, NY
Research Assistant, Fall 2018 [10hr/week]
- Aided Dr. Svetoslava Todorova in the design, programming, and testing of a low-cost, Arduino based environmental sensor to use toward developing a better understanding of harmful algal blooms (HABs)

Syracuse University, Department of Civil and Environmental Engineering, Syracuse, NY
Research Lab Assistant, Fall 2017 [10hr/week]
- Worked in the lab of Dr. David Chandler conducting hydrologic experiments as well as data and lab organization

Leadership/Work Experience

Syracuse University, Syracuse, NY
Teaching Assistant, Spring 2019 [20hr/week]
- Led recitations to review concepts and problem-solving techniques for an introductory course in electromagnetism
- Additional duties included meeting with students to further discuss material as well assisting with grading assignments

Upstate Freshwater Institute, Syracuse, NY
Laboratory Technician Intern, Summer 2018 [40hr/week]
- Analyzed natural water samples for total, dissolved, and soluble reactive phosphorus by photospectrometry
- Gained experience with laboratory quality control, safety, and data management procedures

Le Moyne College, Syracuse NY
President, Engineering Club, Fall 2016 to Spring 2017 [1-2 hr/week]
- Organized programs, trips, and community service events for students interested in engineering
- Prepared and proposed budgets to secure funding for club activities and expenditures

Research Projects

Master's Thesis at Syracuse University: A Feasibility Study of a Low-cost, Large-scale Thermal Treatment method for Human Feces, Fall 2018 to Summer 2019
- Developed numerical and bench-scale models to predict temperatures achieved by feces subject to solar heating

Term Project for Humanitarian Engineering course at SUNY ESF: An Assessment of Various Adsorption Media for Arsenic Removal from Groundwater in the Chitragunj Village near Kolkata, West Bengal, India, Spring 2018

Physics Capstone at Le Moyne College: An Ode to Yelling Fish, Fall 2016
- Investigated analog event horizons via wave blocking in a water channel under the supervision of Dr. David Craig

REU: Syracuse University: Monitoring the Hydrologic Performance of Green Infrastructure in Syracuse, NY, Summer 2016
- Collected hydrologic data with Dr. David Chandler using pressure transducers and temperature sensors *in situ* to understand the response of green infrastructure projects to rainfall events of differing intensity

Applicable Skills

Software: MATLAB, Arduino IDE, Maple, Visual MODFLOW Flex, Microsoft Excel, Microsoft Word, Microsoft PowerPoint
Languages: English (fluent), Spanish (conversational)
Miscellaneous: Physical fitness, student athlete: member of the Le Moyne College cross country and track teams