Proceedings of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2009
August 30 - September 2, 2009, San Diego, California, USA

DETC2009-87461

SYSTEM DYNAMICS THROUGH GRAPHICAL DIMENSIONAL ANALYSIS

Srikanth Tadepalli
Department of Mechanical Engineering
The University of Texas at Austin
Austin, TX 78712
512 – 762 – 0247
tsrikanth@mail.utexas.edu

Kristin L. Wood
Department of Mechanical Engineering
The University of Texas at Austin
Austin, TX 78712
512 – 471 – 0095
wood@mail.utexas.edu

ABSTRACT
Dimensional analysis is a powerful tool used commonly to develop functional relations between variables affecting a physical system. Known for its versatility in incorporating several different energy domains, the method has allowed for dimensional manipulation of diverse parameters. However, the dimensional combinations developed have been, for the most part, strictly confined to static and time invariant systems. Expanding on the process we introduce a graphical approach to dimensionally model dynamic systems for design. Continuing this extension, we present a graphical and topological combination to illustrate the applicability of dimensional analysis as a state equation generation tool. This tool is comparable to conventional differential element analysis in system dynamics, but provides a systematic and visual approach to design modeling. Further, this exposition acts as a learning instrument where complex engineering equations are derived and interpreted through visual perception similar to a block diagram or flow chart. A dynamic system, in the form of a compressed air–water rocket, is also evaluated in this paper for elucidation.

KEYWORDS
Dimensional Analysis, Buckingham π theorem, Directed Flow Graph, System Dynamics, Design Models

INTRODUCTION
Design modeling for system dynamics requires thorough knowledge of the fundamental processes at work and the associated physical laws. Many complex systems are analyzed for design purposes using simplification procedures where components and sub–systems are isolated and modeled. The governing dynamics of each component is influenced by the physics of the problem and the extent to which the particular component affects the response of the system. For systems with multiple components and interactions, a systematic approach needs to be employed to individually model each sub–system that forms the basis for a system level response. Bond graphs [Karnopp et al., 2006] are a graphical visualization scheme for interpreting system dynamics and provide a means for evaluating state equations through a combination of conservation statements and fundamental physical constitutive laws. However, modeling particular interactions, such as heat transfer or thermodynamics, in the bond graph process presupposes intimate knowledge of the topic and is not as intuitively apparent as other traditional techniques. Dimensional analysis used in the conventional form (Buckingham π theorem) [Bridgman, 1931, Tadepalli et al., 2007] also requires skill in matrix manipulation and is constrained by the possibility of having to solve or simplify systems with non–monomial basis [Szirtes, 1998]. A graphical approach with limited mathematical effort mitigates the need to evaluate intricate analytical relations while providing a visual aid for better understanding of the data flow in the system. Such a process is elaborated in this paper supported by the evaluation of a basic dynamic system.

BACKGROUND INFORMATION
Graphs have been widely used in engineering applications with signal flow graphs [Deo, 1974] and state transition diagrams [Johnson et al., 1972] being the most commonly employed visual tools in modeling electrical and control systems. Happ [1971] introduced the concept of illustrating dimensional analysis through the use of directed graphs. Using methods of path inversion and transmittances, Happ [1971] established several engineering identities. However, the basic limitation of these graphs is the inability to model transient or dynamic systems and thus most engineering phenomenon modeled are static, linear or quasi–linear steady state problems. Further, all systems shown by Happ [1971] have simple fundamental laws that are coupled only through variable multiplication and no parameter is added or subtracted which is an inherent requirement for conservation statements that occur frequently in the analysis of mechanical

systems. However, the process pioneered the use of graphs in dimensional analysis.

Shown below are a couple of illustrations showing Happ's approach, focusing on energy transfer in the mechanical and electrical domains. The weights on each edge indicate the value with which the respective node is scaled. The dimensional form of energy $[ML^2T^{-2}]$ is graphically modeled as the product of mass $[M]$ and square of velocity $[V^2]$ (see Figure 1) and equivalence of energy is shown in Figure 2, where two different domains are combined using the scalar equality operator "=" indicating dimensional equivalence.

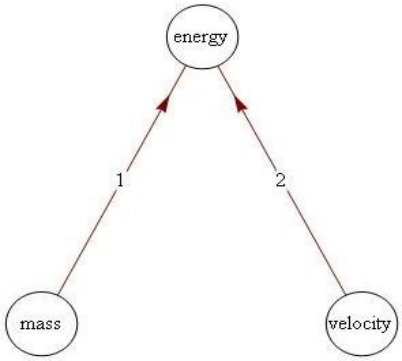


Figure 1. A simple flow graph

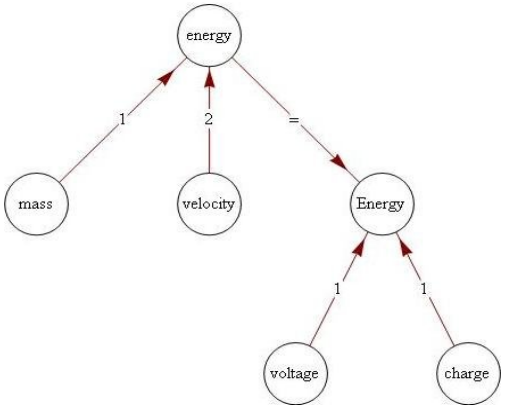


Figure 2. Multiple domain flow graph

We modify this technique to make it more conducive to mechanical engineering design applications, specifically to dynamic systems modeling, by introducing the $\Pi - \Sigma$ approach. Constructing graphs with such engineering relevance is similar to traditional graph structures used in data structures including tree spanning mechanisms and shortest path algorithms. The objectives for using graphs in computing are different from this approach as they are primarily meant for searching, sorting and optimizing while this method focuses mainly on developing finite-edge, complete and dimensionally homogeneous structures associated with physical laws and pertinent to dynamic system behavior.

DEVELOPMENT OF THE GRAPH

A graph is an ordered collection of nodes and edges and hence is defined as -

$$G = \{N, e\} \tag{1}$$

where N are the nodes and e is the number of edges. In engineering systems, nodes represent variables affecting a system and edges capture how these variables interact with each other. This interaction is quantified by the edge weight which is finite for a well–defined relationship (see Figure 3) and indefinite for an unknown or non–existent relationship (see Figure 4). The graph in Figure 3 implies that density is dimensionally equivalent to the ratio of mass and volume, *i.e.*, the product of mass raised to the power of "1" and volume raised to the power of "-1". Hence, the graph is complete and needs no further information to define it. On the contrary Figure 4 emphasizes the idea that no plausible relationship can be established between mass and area without further information relating the two. Hence this graph is incomplete and a broken link is shown with no definitive edge weight.

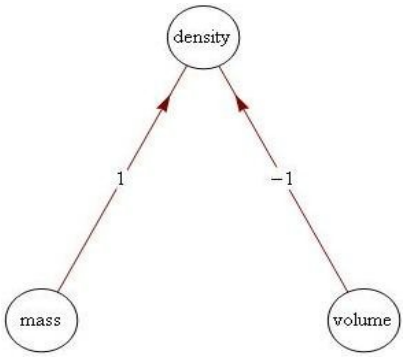


Figure 3. Graph for a known relationship – finite edge weight

Figure 4. Graph for an unknown relationship – indefinite edge weight

The motivation now is to establish a graph structure that is complete and has finite edge weights to capture the true physics of a dynamic system. Any graph with an indefinite edge is discarded as it represents a physical law that is

improbable or unknown[1]. Modifying the relation for the system graph we have,

$$G_{sys} = \{N_{sys}, e_f\} \quad (2)$$

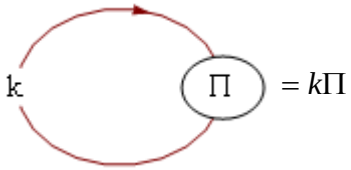
where N_{sys} are the nodes in the system and e_f are the edges with finite edge weights. A node indicating a variable must have dimensions and hence has an associated dimension vector or *D*–vector. The *D*–vector is a row vector with elements equal to the indices of the fundamental vector $[M, L, T, \theta, q]$ (mass – length – time – temperature – charge), corresponding to the basic dimensional analysis variables. A node representing force would have a *D*–vector equal to $[1, 1, -2, 0, 0]$.[2]

With this basic graph structure in mind, we now introduce the following nodes that are developed to simplify the analysis for dynamic system evaluation:

- *Independent node* – A node representing an independent variable.
- *Dependent node* – A node representing a variable that is dependent on at least one other variable, *i.e.*, node.
- *Derived node* – These nodes are the Π and Σ nodes representing the product and the sum respectively of two independent or dependent nodes. These are dependent nodes by default.
- *Differential node* – A node that contains a derivative *w.r.t* a variable.
- *Constant node* – A node with constant numerical value.
- 0 *node* – A node with zero numerical value.
- *F node* – A node indicating a known functional form, *e.g.*, a transcendental function.

These nodes constitute the basic graph nomenclature needed to represent design models for dynamic systems. A set of rules is now needed to develop an approach for deriving graph models. The rules and constraints for the generation of graphs are as follows:

- A Π or a Σ node can have only two inputs.
- A independent or dependent node cannot directly interact with another independent or dependent node. All interactions have to be through a Π or a Σ node to satisfy dimensional analysis properties.
- Every Π and Σ node must represent a valid physical law or principle.
- A graph can have any number of Π and Σ nodes, *i.e.*, there is no limitation on the number of Π and Σ nodes.
- Any two nodes can be used to produce a Π node as long as the resulting Π node is dimensionally correct, physically measurable and has a *D*–vector with at least one element larger in power magnitude than the input nodes.
- A Σ node can be used to combine two nodes of the same dimensional form due to the principle of dimensional homogeneity [Murphy, 1950] and hence cannot have inputs with inconsistent dimensional forms.
- An edge weight of "+" or "-" indicates the positive or negative numerical summation of the concerned nodes respectively.
- A constant node can be used only as a scaling parameter when combined with a Π node. A constant node can never be used with a Σ node.
- A loop indicates that the concerned node is scaled by the constant of the loop only once as shown below.



- A 0 node indicates the culmination or the sink of the graph. The input to the 0 node can have any dimensional form.
- A graph has to have edges with finite edge weight.
- All logarithmic, exponential and transcendental relations can only be combined after reducing to the correct dimensional form – a non-dimensional number. The output of the *F* node has to be a number [Hart, 1995].
- A sub–graph of a system graph must also represent a physical phenomenon.
- The number of sub–graphs must be the same as the number of terms in the conservation statement.
- A node can be combined with itself through a Π and Σ node, consistent with the looping condition, as long as the resulting output is dimensionally correct and physically measurable.
- A system will have one and only one graph to represent its dynamics ensuring uniqueness of the physics and fundamental laws *i.e.*, despite the existence of multiple design models, graphs modeling particular physics laws are unique

Using the definitions and rules developed, we proceed to inductively evaluate a dynamic system that incorporates the functionality and versatility of the $\Pi - \Sigma$ graphical approach.

ILLUSTRATIVE EXAMPLE

The system evaluated as an illustration of the process is a basic toy–rocket assembly (see Figure 5) using water and compressed-air as a charge for propulsion [Otto et al., 2001].

[1] Rules and constraints for graph construction are elaborated later in the paper.

[2] These definitions are used later in the construction of the graph.

The travel of the rocket is governed by the momentum conservation principle where the thrust developed upward is a consequence of the exit of the water jet downward.

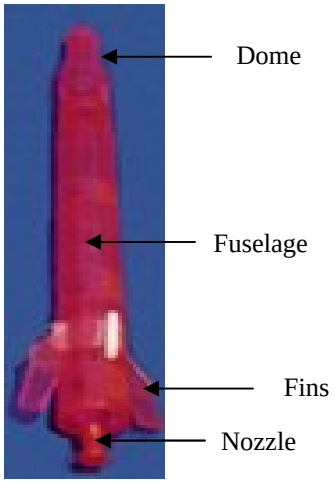


Figure 5. A Simple Water Rocket

We follow a systematic procedure to evaluate the system and develop the state equations. This multi-energetic system serves as a pilot study in showing the feasibility of the $\Pi - \Sigma$ approach as it combines principles from thermodynamics, fluid mechanics and translational dynamics.

Step 1: Identify different domains in the system

Thermodynamics – The isentropic expansion of air causes a pressure differential within the rocket and the outside atmosphere.

Fluid Mechanics – The exit of the jet from the rocket is governed by the continuity equation, momentum equation and Bernoulli's principle.

Translational Dynamics – The motion of the rocket upward is determined by Newton's laws incorporating drag and gravitational or inertial effects.

This is probably the most critical step in the analysis to ensure proper dimensional evaluation of the system. Some technical knowledge and skill is expected of users to identify and understand the principles governing the behavior of the system. The process of recognizing and distinguishing different forces at play in such systems is implicit to the evaluation procedure.

Step 2: Setup conservation statement for each domain

We present the conservation statement for translational dynamics which can be generalized for the entire system. But using a simpler approach where each domain is analyzed for its system parameters offers greater flexibility and ease in modeling. The translational dynamics is governed by,

$$Force_{net} = Thrust\ Developed - Drag - Inertia \qquad (3)$$

Like in the previous step, without any loss of generality, an assumption is made that relations like above can be developed without major concerns, as would be expected of most engineering researchers and design modelers.

Step 3: Evaluate each term by identifying the influencing variables and dimensional analysis

Identification of these variables is sometimes not trivial but careful analysis and understanding of the physics of the problem allows proper evaluation of the system. In this system, since the thrust is caused by the change in *momentum*, the two moving objects causing such change need to be identified first. While the rocket (object 1) motion is the output momentum, the input momentum is the result of rapidly exiting fluid (object 2). The fluid (water) exits because the air within the rocket pushes down on the liquid column forcing discharge. Hence, the thrust developed should be governed by the geometry of the nozzle, the jet velocity at the nozzle and the density of the liquid as these determine flow resistance, flow speed and flow properties respectively. Thus, liquid density (ρ_l), nozzle area (A_n) and the nozzle jet velocity (v_n) are the key variables.

In order to combine these variables, we resort to simple combinations of material and geometry attributes. Consider the variables ρ_l and A_n. While the former is a material specific flow parameter of the liquid used, the latter is a geometry specific variable of the rocket nozzle. Hence, it is fairly obvious that they cannot be directly related. Thus, another variable is needed that couples these two variables *i.e.*, v_n, which combines with each of them independently thereby associating them indirectly. Therefore one of the four identities below must hold true and have physical significance.

$$\begin{aligned} &\{A_n, \rho_l\} \oplus v_n \\ &\text{where } \oplus \text{ is } \{+, -, \times, \div\} \end{aligned} \qquad (4)$$

Note that addition and subtraction would be dimensionally incorrect due to disparity in dimensions between the parameters. Thus, the set of possible operators reduces to $\{\times, \div\}$. The next challenge is to understand if any parameter needs to be scaled to a power *i.e.*,

$$\begin{aligned} &\{A_n, \rho_l\} \oplus v_n^k \\ &\text{where } \oplus \text{ is } \{\times \text{ for } k = n, \div \text{for } k = -n\}, \\ &n \in N \end{aligned} \qquad (5)$$

This issue is resolved using the rules and constraints developed for construction of the graph. Let $k = 1$. This

generates $A_n v_n$ and $\rho_l v_n$. For k = -1, we get $\frac{A_n}{v_n}$ and $\frac{\rho_l}{v_n}$. The latter two are unknown physical quantities as is the second relation of the first set. However, the first relation of the first set, $A_n v_n$, bears dimensional and measurable features of a known quantity, that of volume flow rate. Thus for k = 1, we have identified and isolated a valid physical parameter consistent with the rules and hence need no more iterations of k. Proceeding with the construction of the entire graph using similar logic, we have (see Figure 6),

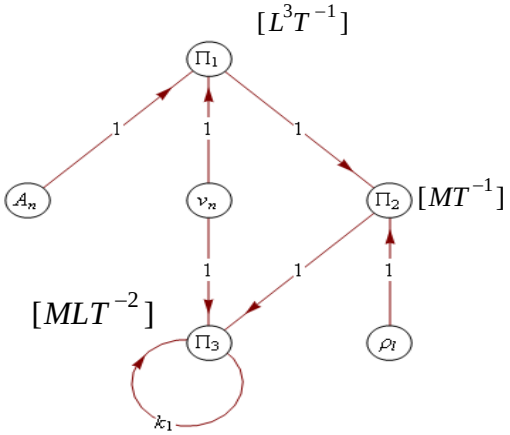


Figure 6. Flow graph for thrust

The graph thus represents thrust with dimensions of force given by,

$$Thrust\ Developed = k_1 \rho_l v_n^2 A_n \quad (6)$$

Notice that all edge weights are numerically finite, every influencing variable has been incorporated and each Π group has a D–vector greater than or equal to the preceding nodes. Each Π group represents a dimensionally sound measurable quantity with Π_1 accounting for volume flow rate, Π_2 indicating mass flow rate and Π_3 signifying force. Notice also that no other combination of the variables produces a measurable quantity thus validating the uniqueness property of the graph. Note also that the final node Π_3 is scaled by a constant since dimensional analysis does not yield any constants in an equation. This is a limitation of using dimensional analysis as the values of the constants needs to be established through experimentation as shown later. Similarly, we now evaluate the drag experienced by the rocket and the inertial effects encountered in its flight path. The parameters that affect drag are co–efficient of drag (C_d), velocity of the rocket (v_r), air density (ρ_a) and the rocket area (A_r). The only two factors affecting inertial effects are the total instantaneous mass of the rocket (m) and acceleration due to gravity (g). Setting up the graphs accordingly, we have (see Figures 7 and 8),

$$Drag = k_3 C_d \rho_a v_r^2 A_r \quad (7)$$

$$Inertia = k_2 mg \quad (8)$$

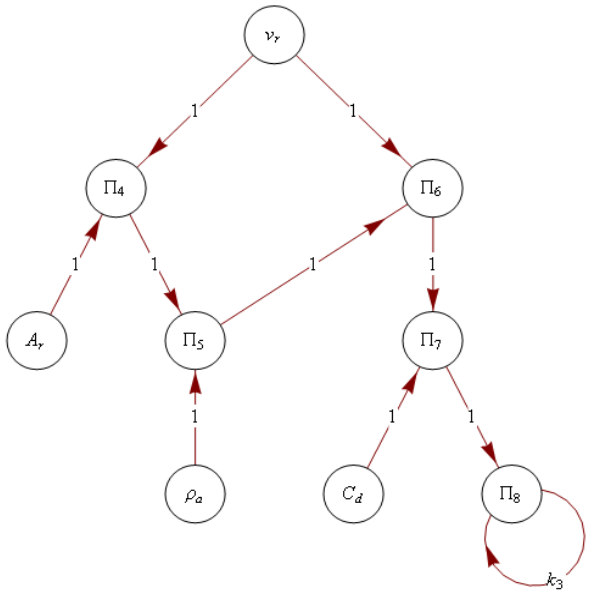


Figure 7. Flow graph for drag

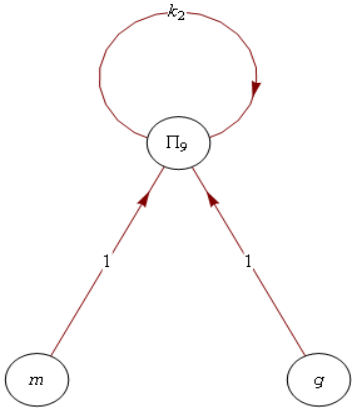


Figure 8. Flow graph for gravity

Step 4: Combine graphs if all terms are evaluated in the conservation statement

Since we have accounted for all terms in the conservation statement, we can now combine the sub–graphs to generate the parameter graph for translational dynamics. Notice that all end Π groups in each sub–graph *i.e.*, Π_3, Π_8 and Π_9 have dimensions of force and thus the principle of homogeneity is satisfied. Combining these forms in the relation $\frac{\Pi_i \Pi_j}{\Pi_k}$ still generates an output form that is dimensionally equivalent to

force but violates the conservation statement (see Equation (3)) and hence is not valid. Therefore the only plausible relation that is both dimensionally accurate and valid from a conservation standpoint has to have a relation given by,

$$Force_{net} = \Pi_3 - \Pi_8 - \Pi_9 \tag{9}$$

We invoke the Σ nodes using the "+ -" convention to combine the Π nodes for the parameter graph of translational dynamics as shown below (see Figure 9). The net parameter graph indicates the net force upward which is the combination of mass and the acceleration upward and hence the following graph is generated. Thus Π_{10} in the graph below indicates the net force upward given by $m\frac{dv_r}{dt}$ which is the thrust Π_3 minus the total resistance Σ_1 (drag + gravity).

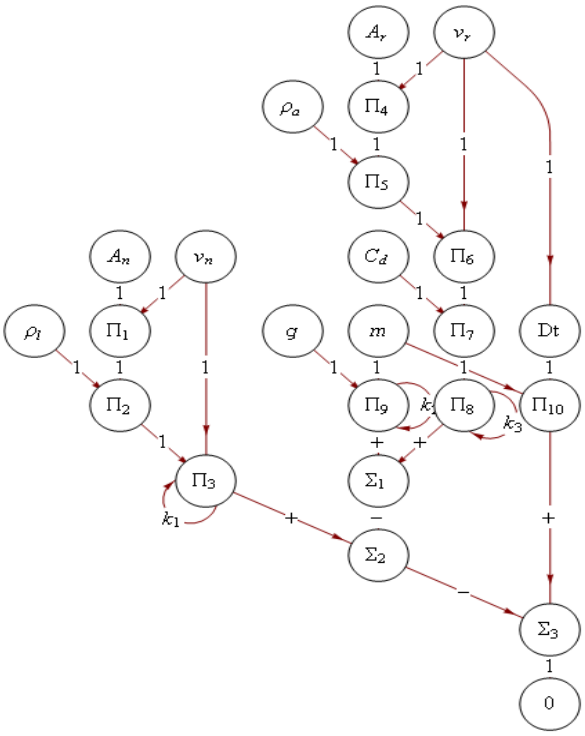


Figure 9. Flow graph for translational dynamics

Notice that the node v_r is associated with the differential node *Dt* with a unit scale on the bond indicating a first order derivative. Thus, the scale signifies the order of differentiation of the variable node with respect to the differential node. The generated flow graph thus represents the analytical relation given by,

$$m\frac{dv_r}{dt} = k_1\rho_l v_n^2 A_n - k_2 mg - k_3 C_d \rho_a v_r^2 A_r \tag{10}$$

The flow graph for translational dynamics is thus complete, dimensionally homogeneous and has finite edge weights. Equation (10) thus represents a basic first-order, non–linear model for the translational dynamics of the system. Further, notice that all variables are constants except for v_n, a parameter we have introduced for completeness, Thus, v_n needs to have a state equation too to define its behavior. We thus seek state equations to *completely* model the system and every time a new variable is introduced that is neither a constant nor a known parameter, we refine the equation set to update a new state equation iterating till all the domains listed in Step 1 are captured.

Step 5: Generate graphs for each domain

Using similar procedures for the Bernoulli's, isentropic expansion and continuity equations we generate the following parameter graphs using the variables liquid density (ρ_l), nozzle area (A_n), total instantaneous mass of the rocket (m), nozzle jet velocity (v_n), Co–efficient of nozzle (C_N), mass of water (m_l), pressure difference (ΔP), Co–efficient of expansion ($C_{isentropic}$), volume of rocket (V_r) and the gas constant (K) (see Figure 10).

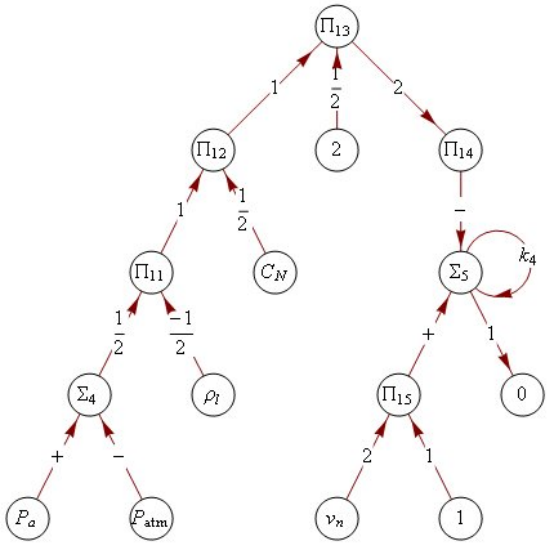


Figure 10. Flow graph for Bernoulli's equation

This graph thus gives the relation,

$$v_n^2 = \frac{k_4 C_N (P_a - P_{atm})}{\rho_l} \tag{11}$$

where we have now introduced the internal air pressure P_a as an unknown parameter. The rest of the equation has terms that

have been used before or are known constants. To model the variation in pressure, we invoke the next state equation captured by isentropic expansion (see Figure 11) given by,

$$P_a\left[V_r - \frac{m_l}{\rho_l}\right]^K = k_5 C_{isentropic} \tag{12}$$

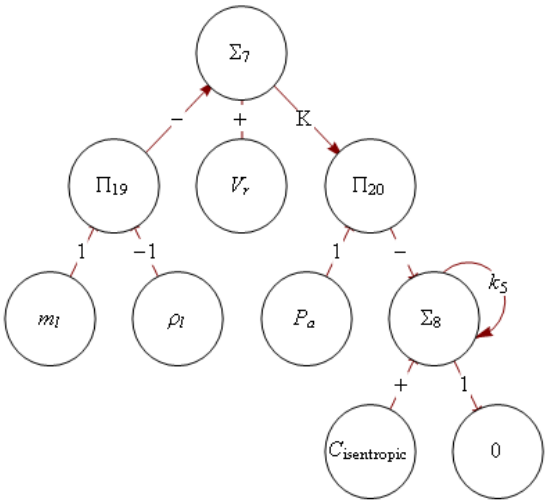


Figure 11. Flow graph for isentropic expansion

thus introducing an additional unknown parameter m_l, which is modeled using the continuity equation (see Figure 12) as,

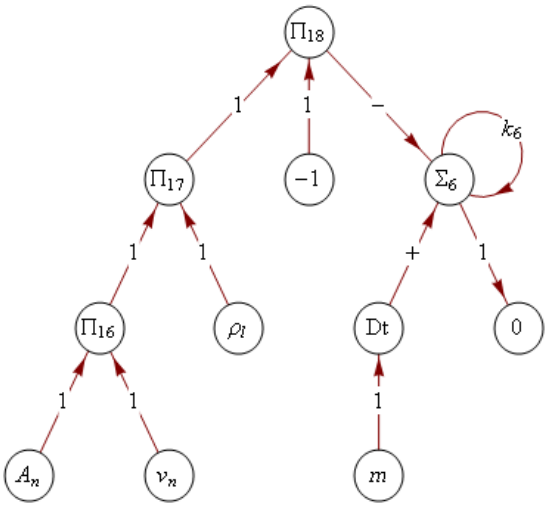


Figure 12. Flow graph for continuity

$$\frac{dm}{dt} = -k_6 \rho_l A_n v_n \tag{13}$$

where m is total instantaneous mass of the rocket. At this point, we have neither introduced a new variable nor have any unknown parameters in the latest state equation suggesting that all domains have been captured. Thus, the number of state equations spans the list of domains. Hence, the state equations given by the dimensional graphs are,

$$\begin{aligned}
&\frac{dv_r}{dt} = k_1\left(\frac{\rho_l A_n v_n^2}{m}\right) - k_2 g - k_3\left(\frac{C_d \rho_a A_r v_r^2}{m}\right)\\
&v_n^2 = \frac{k_4 C_N (P_a - P_{atm})}{\rho_l}\\
&P_a\left[V_r - \frac{m_l}{\rho_l}\right]^K = k_5 C_{isentropic}\\
&\frac{dm}{dt} = -k_6 \rho_l A_n v_n
\end{aligned} \tag{14}$$

Step 6: Evaluate constants in the relationships

Evaluation of constants in the state equations is performed by grouping numerical simulation and experimental verification. State equations have six different constants whose values need to be ascertained. The experimental procedure provides values for height achieved alone, which is the only available benchmark. This height is the net sum of the individual heights that the rocket attains *i.e.*, the height achieved till the system exhaust all charge and the height achieved thereafter. While the equations above capture the first event, the latter motion can be modeled respectively using,

$$\frac{dv_r}{dt} = -k_2 g - k_3\left(\frac{C_d \rho_a A_r v_r^2}{m}\right) \tag{15}$$

and

$$\frac{dv_r}{dt} = -g - \left(\frac{C_d \rho_a A_r v_r^2}{2m}\right) \tag{16}$$

This allows for capturing the maximum height reached by the rocket that includes the two individual events of flight during water drain and subsequent flight. The experimental setup is shown below (see Figure 13) and results are summarized thereafter (see Table 1).

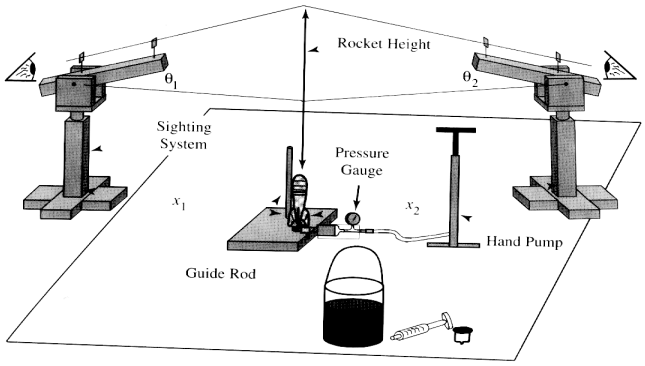


Figure 13. Experimental Setup for the Toy Rocket

Excel solver is used to minimize error from numerical evaluation using conjugate search and central derivatives (see Figure 14). An initial guess for all six constants commences process of estimation. Bear in mind that choice of initial guess greatly affects convergence and hence an educated guess needs to be employed. Further, optimality is dependent on inputs as well and hence, the optimal combination varies as values of inputs differ (see Table 2). The system is simulated for known inputs to establish numerical height achieved. For an initial volume of 50% water pressurized to 4 atm gage (see Table 3), simulations are run up until the system reaches its net height which is evaluated to be 17.52 m as compared to an experimental value of 17.92 m causing an error of 0.40 m. This error, specific to an initial volume of 50% water and pressure of 4 atm gage, is minimized using the solver technique to obtain sub-optimal values for constants as shown below.

Table 1. Experimental Values

Experimental Values
Net Height (m)
14.14
17.92
19.34
15.86
12.27
7.03

Step 1: Consider system of equations. Set all constants to an initial guess. Go to step 2.
Step 2: Estimate height with assumed values of constants for one test condition (say %vol of water = 0.5). Go to step 3.
Step 3: Compare simulated numerical height with experimental value for the test condition. Estimate error and go to step 4.
Step 4: Setup an optimization scheme to minimize error for the particular test condition by altering values of constants. Go to step 5.
Step 5: Obtain a sub-optimal set of values for the specific test condition. Go to Step 6.
Step 6: Using the sub-optimal values, compare for different input conditions to generate individual errors for each of the test condition. Go to Step 7.
Step 7: Alter the sub-optimal values of the constants such that each individual error ***and*** the net root mean square of the individual errors are minimized to obtain global optimal values.

Hence, the optimization problem can be mathematically defined as,

$$\text{find } \{k_i\}_{i=1}^{6} \ni \varepsilon_i \text{ and } \sqrt{\frac{\sum_{i=1}^{6} \varepsilon_i^2}{6}} \text{ are minimized } \forall\, i \in [1,6]$$

where

$$\varepsilon_i = \left|\text{num}_i - \exp_i\right| \forall\, i \in [1,6]$$

In essence, the procedure iterates till an optimal combination of constants is obtained where each individual error ***and*** the net RMS error across all six readings are minimized. Once the optimal set of constants is established (see Table 4), the values are incorporated into the state equations and simulated for varying levels of %vol of water in the system. Since constants are specific to a particular condition, errors are still generated but remain convergent and below the acceptable error tolerance when inputs change (see Figure 14). The algorithm is put to test with the set of initial values for all parameters and constants as shown below [Otto et al., 2001]. All units are given in SI system.

Table 2. Input Parameters and Values

Input	Parameter	Initial value
Area of Jet	A_j	0.174E-04
C/S Area of Rocket	A_r	5E-04
Volume of Rocket	V_r	75E-06
Mass of rocket	m_r	0.0165
Nozzle Coefficient	C_N	1.0
Drag Coefficient	C_d	0.1
Pressure of Air	P_a	451.325K
Volume of Water	V_a	37.5E-06
Density of Air	ρ_a	1.293
Density of Water	ρ_w	1000
Acceleration due to gravity	g	9.81
Expansion Coefficient	K	1.40
Atmospheric Pressure	P_o	101.325 K

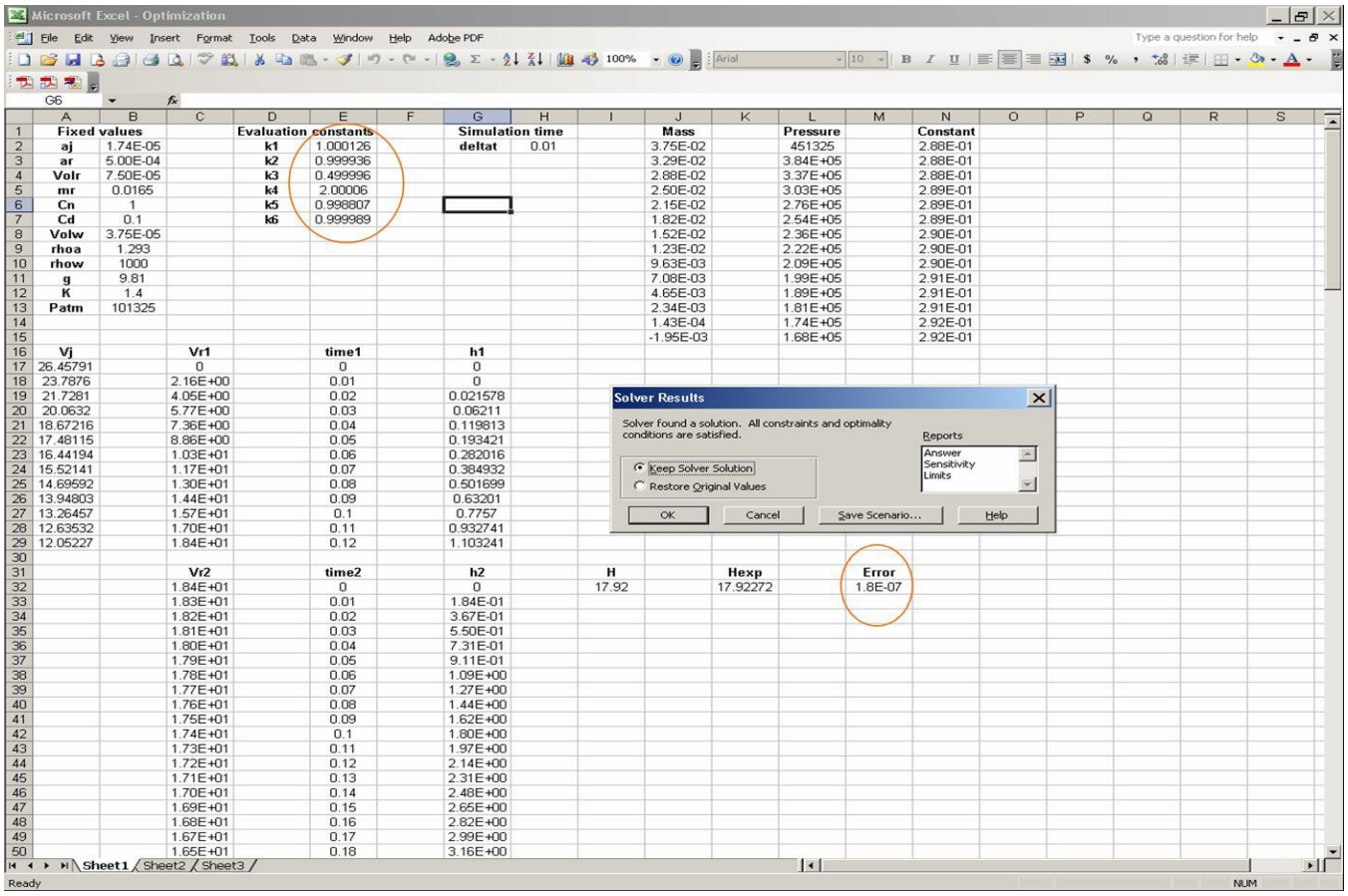


Figure 14. Optimization using Excel solver

Table 3. Inputs to the Experimental Setup

%Volume of water (m^3)	Air Pressure (N/m^2)
0.6	4.51E+05
0.5	
0.4	
0.3	
0.2	
0.1	

Table 4. Estimated values of the Constants

Constant	Estimated Value
k_1	1.00
k_2	0.99
k_3	0.49
k_4	2.00
k_5	0.99
k_6	0.99

Incorporating the constants into the state equations generates the mathematical model given by,

$$\frac{dv_r}{dt} = \left(\frac{\rho_l A_n v_n^2}{m}\right) - 0.99\ g - 0.49\left(\frac{C_d \rho_a A_r v_r^2}{m}\right)$$

$$v_n^2 = \frac{2C_N(P_a - P_{atm})}{\rho_l}$$

$$P_a\left[V_r - \frac{m_l}{\rho_l}\right]^K = 0.99 C_{isentropic} \qquad (17)$$

$$\frac{dm}{dt} = -0.99\rho_l A_n v_n$$

We now compare the developed state equations to the traditional state equations given below [Otto, 2001].

$$\frac{dv_r}{dt} = \left(\frac{\rho_l A_n v_n^2}{m}\right) - g - \left(\frac{C_d \rho_a A_r v_r^2}{2m}\right)$$
$$v_n^2 = \frac{2C_N (P_a - P_{atm})}{\rho_l} \qquad (18)$$
$$P_a\left[V_r - \frac{m_l}{\rho_l}\right]^K = C_{isentropic}$$
$$\frac{dm}{dt} = -\rho_l A_n v_n$$

Notice that apart from the values of constants, the two equation sets bear absolute similarity in their respective terms.

To compare the graphical approach, based on dimensional analysis, to classical modeling, we derive the translational dynamics equation using conventional control volume–control surface analysis (see Figure 15).

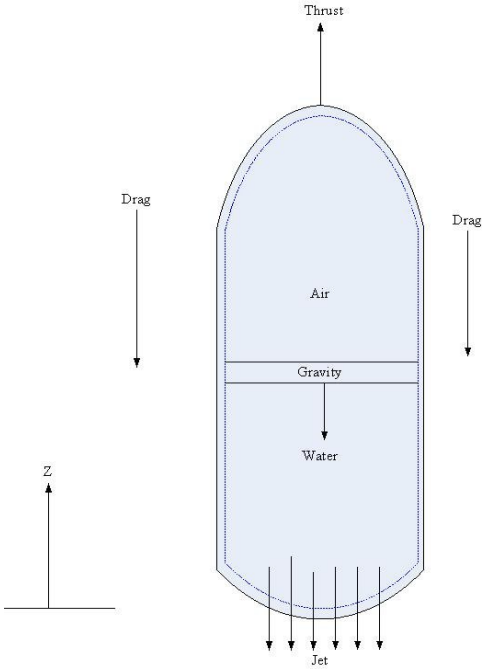


Figure 15. Control volume for the toy rocket

Conserving momentum, we have,

$$\sum \overrightarrow{F_{ext}} = \frac{d\vec{P}}{dt} = \sum_{CS} \vec{v}\rho\vec{v} \bullet \vec{A} + \frac{d}{dt}\int_{CV} \vec{v}\rho \bullet d\vec{V} \qquad (19)$$

$\vec{v}$ being the velocity vector, $\vec{A}$ being the area vector and $\vec{V}$ being the volume vector. Since the forces acting on the system are aligned in the *z*–axis (vertical direction) alone,

$$\sum F_z = \sum_{CS} \overrightarrow{v_n}\rho_l \overrightarrow{v_2} \bullet \overrightarrow{A_n} + \frac{d}{dt}\int_{CV} \overrightarrow{v_n}\rho_l \bullet d\vec{V} \qquad (20)$$

where $\vec{v}_n$ is the jet velocity vector *w.r.t* the vertical reference frame and $\vec{v}_2$ is the velocity vector *w.r.t* the control volume. This implies that

$$\vec{v}_n = \vec{v}_2 + \vec{v}_r \qquad (21)$$

where $\vec{v}_r$ is the rocket velocity vector. Substituting equation (21) in equation (20), we have,

$$\sum F_z = (v_r - v_n)\rho_l v_n A_n + \frac{d}{dt}\int_{CV} v_2 \rho_l dV + \frac{d}{dt}\int_{CV} v_r \rho_l dV \qquad (22)$$

Since the velocity vector *w.r.t* the control volume, $\vec{v}_2$, does not change with time internal to the control volume,

$$\frac{d}{dt}\int_{CV} v_2 \rho_l dV = 0 \qquad (23)$$

Further since $\int_{CV} \rho_l dV = m$, we have,

$$\sum F_z = (v_r - v_n)\rho_l v_n A_n + \frac{d}{dt}(m v_r) \qquad (24)$$

or,

$$\sum F_z = (v_r - v_n)\rho_l v_n A_n + m\frac{dv_r}{dt} + v_r \frac{dm}{dt} \qquad (25)$$

From the continuity relation,

$$\frac{dm}{dt} = -\rho_l A_n v_n \qquad (26)$$

Substituting equation (26) in equation (25), we have,

$$\sum F_z = (v_r - v_n)\rho_l v_n A_n + m\frac{dv_r}{dt} - \rho_l A_n v_n v_r \qquad (27)$$

Simplifying,

$$\sum F_z = -\rho_l v_n^2 A_n + m\frac{dv_r}{dt} \qquad (28)$$

Recognizing the fact that the only two external forces acting on the system are the weight and drag resistance, we have,

$$\sum F_z = -mg - \frac{1}{2}\rho_a C_d A_r v_r^2 = -\rho_l v_n^2 A_n + m\frac{dv_r}{dt} \qquad (29)$$

Reorganizing the equation,

$$m\frac{dv_r}{dt} = \rho_l v_n^2 A_n - mg - \frac{1}{2}\rho_a C_d A_r v_r^2 \qquad (30)$$

Equations (10) and (30) are identical except the derivation process is significantly more intricate in the conventional approach. The graphical approach, on the other hand, offers a

visual perspective where the transition from one term to the other is intuitive and unequivocal. Thus, an entire mathematical development can be replaced by a single dimensional graph. On the downside, the conventional method does account for constants in the equation unlike the dimensional approach. However, this presents an opportunity to combine experimentation and optimization. This approach is thus comparable to other popular visual modes of design modeling and analysis including Finite Element Methods (FEM) and Computer Aided Design (CAD) which also use non–linear solvers that are typically integrated with the software.

PRACTICAL IMPLEMENTATIONS

The development of design models for system dynamics using graphs has allowed for state equation generation using only a set of experiments and basic dimensional analysis results. The constants developed in the equation remain impervious to magnitude changes in parameters (geometric or otherwise) indicating the uniqueness of the model and the underlying physical phenomena. Hence, this is a procedure where a design model is derived from a limited set of data points and experiments. Repetition and reevaluation of constants is not necessary as they are invariant to changes in magnitudes in parameters and are thus dependent on the system and its physics alone rather than the numerical magnitude that each parameter attains. To validate this argument, pressure is changed and in each case the experimental and numerical solutions are compared where the numerical solutions incorporate the derived constants. The two solutions are shown below for varying levels of pressure and %water level fixed at 0.4 (see Figure 16). Notice that the differences between the numerical values are marginal implying robustness in the developed design model.

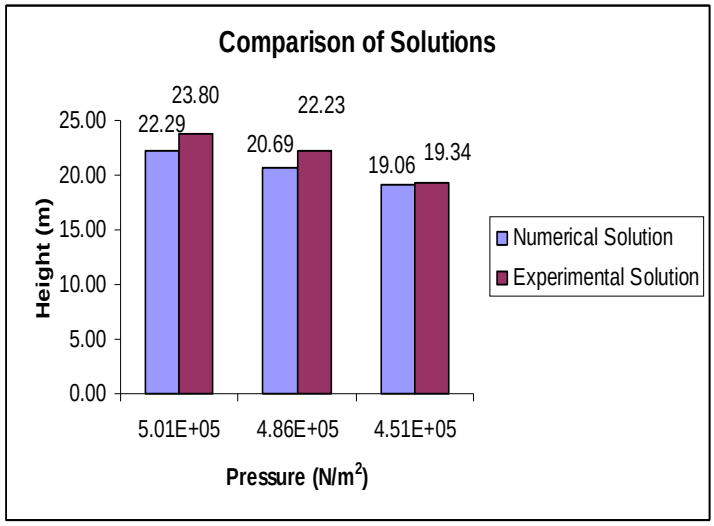


Figure 16. Comparison of numerical and experimental solutions

Having established the working procedure of the graphical dimensional analysis method and illustrated an example of a dynamic system, a concise flow chart (see Figure 17) is offered below that encapsulates the development.

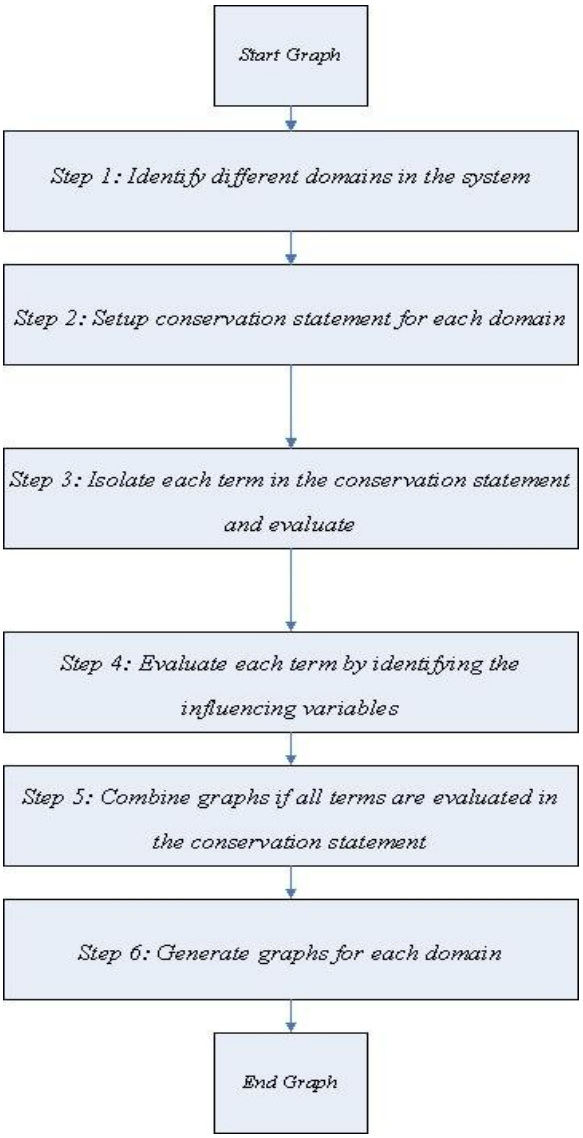


Figure 17. Flow chart for graphical dimensional analysis

CONCLUSIONS AND DISCUSSION

Graphs provide visual understanding of engineering principles by simple pictorial representations of complex analytical relations. Combined with dimensional analysis, they offer an alternate evaluation procedure for system dynamics and thus provide diagrammatic insight into the working processes of engineering phenomena. As a learning tool, graphs and dimensional analysis illustrate the applications of engineering parameters in scientific study of physical systems. It is also widely acknowledged from cognitive and psychological studies that pictures convey more information than words. Thus, a student being introduced to dynamic system evaluation is inclined to assimilate graphical layout rather than

conventional text derivations. Such hypothesis though needs student surveys and statistical corroboration which would lend credibility to this effort.

System graphs can also be generated by combining all individual graphs so that a global scheme is developed where all influencing parameters are accounted for and their interactions with each other are quantified. Such a pictorial representation allows the designer to traverse the graph for nodes which can be tweaked, like geometric constants, and the effect such a manipulation causes in the graph in terms of functional dependence. Since each Π node represents a variable that can be controlled, the global graph presents an opportunity of identifying and isolating key controllable variables so that system response can be adjusted accordingly. Note that a system modeled using *n*–differential equations would still result in a single system graph with several interactions and such a representation allows for referencing and combining all influencing variables in a single plot unlike the traditional approach. As a potential extension, the system graph may be coupled with simulator programs such as LabView to visually model and simulate the response of a mechanical system.

The methodology developed in this technique imparts initial impetus to the process of graphical dimensional analysis. While the procedure employed many rules and regulations in graph construction, continued exposure and usage would reduce difficulty in execution when more potential engineering design models are realized.

References

[1] Baker, W. E., Westine, P. S., Dodge, F. T., 1991, *Similarity Methods in Engineering Dynamics: Theory and Practice of Scale Modeling*, Elsevier.

[2] Barr, D. I. H., 1982, *A Survey of Procedures for Dimensional Analysis*, International Journal of Mechanical Engineering Education, 11/3, pp 147 – 159.

[3] Bridgman, P. W., *Dimensional Analysis*, 1931, Yale University Press, New Haven.

[4] David, F. W., Nolle, H., 1982, *Experimental Modelling in Engineering*, ButterWorths.

[5] Deo N., 1974, *Graph Theory with Applications to Engineering and Computer Science*, Prentice Hall.

[6] Edited by Klamkin, Murray S., 1987, *Mathematical Modelling: Classroom Notes In Applied Mathematics*, SIAM.

[7] Happ W.W., 1971, *Dimensional Analysis via Directed Graphs*, Journal of the Franklin Institute, 292/1 July, pp 527 – 533.

[8] Happ, W.W., 1967, *Computer-Oriented Procedures for Dimensional Analysis*, Journal of Applied Physics, 38/10 September, pp 3918 – 3926.

[9] Hart, G.W., 1995, *Multidimensional Analysis*, Springer – Verlag.

[10] Johnson D.E., Johnson J.R., 1972, *Graph Theory with Engineering Applications*, The Ronald Press Company, New York.

[11] Karnopp, D.C., Margolis, D.L., Rosenberg, R.C., 2006, 4th ed., *System Dynamics: Modeling and Simulation of Mechatronic Systems*, John Wiley & Sons, Hoboken, NJ.

[12] Langhaar, H. L., 1951, *Dimensional Analysis and Theory of Models*, John Wiley & Sons, New York.

[13] Moon, P., Spencer D.E., 1949, *A Modern Approach to "Dimensions"*, Journal of the Franklin Institute, 248/1483-1488 (July – December), pp 495 – 521.

[14] Murphy, G., 1950, *Similitude in Engineering*, The Ronald Press Company, New York.

[15] Otto, K. N., Wood, K.L., 2001, *Product Design – Techniques in Reverse Engineering and New Product Development*,, Prentice Hall, Upper Saddle River NJ.

[16] Rayleigh, 1915, *The Principle of Similitude*, Nature, 2368/95, 66 – 68.

[17] Sharp, J.J., 1975, *Application of Dimensional Reasoning to Thermal Systems*, Journal of the Franklin Institute, 299/3 March, pp 191 – 197.

[18] Skoglund, V.J., 1967, *Similitude – Theory and Applications*, International Textbook Company, PA.

[19] Szirtes, T., 1998, *Applied Dimensional Analysis and Modeling*, McGraw – Hill, New York.

[20] Szücs, E., 1980, *Similitude and Modelling*, Elsevier.

[21] Tadepalli, S., Wood, K.L., 2007, *Reduction of Non-Monomial Basis in the Dimensional Analysis of a Dynamic System*, 33rd Design Automation Conference, ASME – DETC/CIE International Conference Proceedings, Las Vegas, NV, September 4 – 7.

[22] Taylor, E. S., 1974 *Dimensional Analysis for Engineers*, Clarendon Press, Oxford.

[23] Van Driest, E.R., 1946, *On Dimensional Analysis and the Presentation of Data in Fluid-flow Problems*, Journal of applied mechanics, 13, pp A34 – A40.