The Dimensional Analysis Toolbox for Matlab

Steffen Brückner
Institut für Statik und Dynamik der Luft- und Raumfahrtkonstruktionen
Universität Stuttgart, Paffenwaldring 27, D-70569 Stuttgart
brueckner@isd.uni-stuttgart.de

Abstract.
The Dimensional Analysis Toolbox for Matlab/1/ (DA-TB) is a freely available toolbox for the engineering analysis software Matlab. The DA-TB implements symbolic dimensional analysis as well as numeric data transformation subroutines and features a graphical user interface to assist the user in the basic steps of dimensional analysis. This paper describes the basic functionality of the DA-TB, the implementation, and gives to examples of the application of the DA-TB to symbolic dimensional analysis and numeric data transformation.

1. The Matlab Framework and the DA-TB

In the domain of engineering data analysis and modeling, the Matlab (MATrix LABoratory) software/2/ is a well-known numerical mathematics tool. The Matlab framework consists of a base Matlab kernel, implementing I/O-functions, data structures, flow control, and additionally numerous toolboxes built onto this kernel functionality. Toolboxes are available for a broad range of applications, from signal processing and control system design, up to partial-differential equation analysis, neural networks, or fuzzy logic. Symbolic computation features are available to the Matlab user via the Symbolic Math toolbox which includes a Maple kernel.
The DA-TB described here is new, freely available, extension to Matlab which allows for symbolic dimensional analysis, supported with a graphical user interface, as well as numeric data transformation. Using the DA-TB, the researcher or engineer can profit from computer-assisted dimensional analysis and data transformation in the same framework which is used for modeling and simulation. This eliminates the necessity of transforming symbolic dimensionless groups or data from one application to another or between different platforms.

In the following text, first the fundamentals of dimensional analysis are shown. Then two examples of the application of the DA-TB including data transformation are presented.

2. Dimensional Analysis

Buckingham's Pi-Theorem/3/ states that for each dimensional homogeneous and complete relationship *f* of *n* physical variables x_i

$$f(x_1,\ldots,x_n) = 0 \quad (1)$$

there exists a corresponding relationship

$$F(\pi_1,\ldots,\pi_m) = 0 \quad (2)$$

of only m £ n dimensionless groups

$$_j = \prod_{k=1}^{r} x_k^{-C_{jk}} \prod_{i=r+1}^{n} x_i^{-D_{ji}} \;\; ; D_{ji}, C_{jk} \in \mathbb{R} \; ; m = n - r \tag{3}$$

where r denotes the rank of the dimensional matrix, i.e. the number of independent dimensions.

The dimensional matrix can be established from the knowledge of the problems' relevance list, which is the list of all relevant parameters and their respective dimensions. The columns of the dimensional matrix correspond to the parameters while the rows correspond to the value of the dimension exponent of the variable. The elements of the dimensional matrix are therefore the exponents of the dimensions of all the relevant parameters.

$$\underline{M} = [\underline{B} \; \underline{A}] = \begin{array}{c|ccc:ccc} & x_{r+1} & \cdots & x_n & x_1 & \cdots & x_r \\ \hline \mathrm{Dim}_1 & m_{(r+1)1} & & m_{n1} & m_{11} & \cdots & m_{r1} \\ \vdots & \vdots & \underline{B} & \vdots & \vdots & \underline{A} & \vdots \\ \mathrm{Dim}_k & m_{(r+1)k} & & m_{nk} & m_{1k} & \cdots & m_{rk} \end{array} \tag{4}$$

The dimensional matrix ***M***, shown in eq. (4) on the left, is formed by the relevant parameters x_i as the columns and the corresponding dimensional exponents m_{ij} as the rows of the matrix. The dimensional matrix ***M*** can be partitioned into the sub matrix ***A*** (k x r) which contains the r = *rank(**M**)* base variables and the sub matrix ***B*** (k x $(n$-$r)$) which contains the n-r dependent variables. The dimensional matrix can be extended to a dimensional set which contains the additional sub matrices ***C*** and ***D***.

$$\begin{array}{c|c} & x_i \\ \hline \begin{array}{c} \mathrm{Dim}_i \\ p_j \end{array} & \left[\begin{array}{c:c} \underline{B} & \underline{A} \\ \hdashline \underline{D} & \underline{C} \end{array}\right] \end{array} \tag{5}$$

In this dimensional set the sub matrix ***D*** (*(n-r)* x *(n-r)*) can be nearly freely chosen with the only limitation of ***D*** being regular. The sub matrix ***C*** (*(n-r)* x *r*)is the determined by the relation/7/

$$C = -D \cdot (A^{-1}B)^T \tag{6}$$

The dimensionless groups are then determined as

$$p_j = \prod_{i=1}^{r} x_i^{C_{ji}} \prod_{l=r+1}^{n} x_l^{D_{jl}} \tag{7}$$

These dimensionless groups form a minimal set of parameters for the given problem. Due to this property, data plots in dimensionless groups are more compact than plots in dimensional variables and often allow better visualization of physical effects. Therefore, this property is mainly used in aerodynamics and thermodynamics. Also, these dimensionless groups, if held constant, account for physical similarity. If the values of all dimensionless groups for a given problem are equal for a specimen and a scale model, physical similarity is fulfilled. If only some dimensionless groups can be held constant, e.g. due to restrictions in material or manufacturing, this is called partial similarity.

3. Implementation of the DA-TB

The DA-TB is based on two data structures. The `relevance list` data structure contains the relevance list whereas the `piset` data structure contains the dimensional set.

The relevance list data structure containing the relevant variables with the parameters *variable name* and *dimension*, and *factor* is a prerequisite to calculate the dimensional set. The *dimension* parameter in the relevance list is a vector of the dimensions' exponents of the basic SI-units Kilogram, Meter, Second, Kelvin, Ampère, Candela, and Mol. Other units can be transformed into SI basic units using the function `unit2si` which determines the dimensional representation in SI basic units as well as a conversion factor *factor*.

The set of dimensionless groups can be determined from the relevance list and an appropriate choice of base variables using the function `diman` which implements the above described method of dimensional analysis/7/ using base Matlab functionality, such as numeric matrix inversion and matrix multiplication. The `diman` command returns a `piset` structure containing the dimensional set.

`p.A`	A sub matrix
`p.B`	B sub matrix
`p.C`	C sub matrix
`p.D`	D sub matrix
`p.order`	order of the variables from the relevance list in the dimensional matrix (depends on the choice of base variables)
`p.Name`	variable names in the order they appear in the dimensional matrix

Fig. 1: The `piset` structure for dimensional sets

The dimensionless groups determined by `diman` can either be printed to screen or a file using the `pretty` function in ASCII text and in LaTeX format using the `latex` command. These two commands already exist in the Symbolic Math toolbox and are overloaded by the Dimensional Analysis toolbox for dimensional sets as input parameters. To ensure this functionality the DA-TB should appear earlier in the Matlab search path than the Symbolic Math Toolbox. `pretty` and `latex` use the functionality of the Symbolic Math TB, if available, for formatting the output. If the Symbolic Math TB is not installed, internal routines of the DA-TB are used. Unfortunately, due to limitations in

Matlab, it cannot be checked if there is an open license of the Symbolic Math TB if installed in a floating network license environment. In this case, small modifications to the `latex.m` and `pretty.m` source code may be necessary to avoid error messages.

Data can be transformed first to basic SI units using `data2si` and then be transformed to the dimensionless representation using `dtrans`.

The functionality of the `datool` GUI has been built as a basic Matlab GUI using Matlab R12.1. Due to structural changes in GUI programming in Matlab, at least Matlab R12 is needed to access the GUI. Since Matlab only offers a basic array editor that is not capable of editing cell arrays with mixed character and numeric data elements an external ActiveX plugin, "LiteGrid", is used to edit the dimensional matrix and the D-matrix. The drawback with this implementation is that ActiveX plugins require a Windows operating system. Using another OS, the array editor functionality cannot be used.

The DA-TB comes with an installation routine based on Matlab that runs on all platforms and automatically installs the DA-TB. The installation routine allows a single user installation in a local directory, as well as system-wide installation. For the regular, system-wide ("all users") installation, the toolbox has to be installed with administrative privileges.

In the following, two examples for dimensional analysis and data transformation using the Dimensional Analysis Toolbox for Matlab are given.

4. Example: Drag force on a sphere

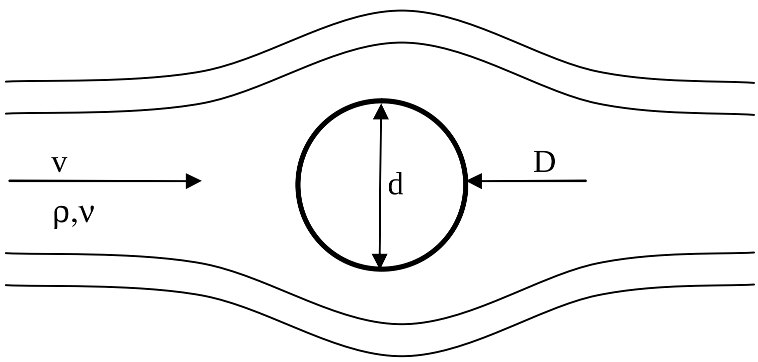

Fig. 2: Drag force on a sphere

In this simple fluid mechanics example, the drag force on a sphere is examined. This drag force is mainly influenced by the flow velocity, the fluid density and viscosity, and the sphere diameter as has been shown by measurements of Prandtl/8/. The list of relevant parameters is therefore given by

drag force	D	[N]
flow velocity	v	[m/s]
fluid viscosity	ν	[m^2/s]
fluid density	ρ	[kg/m^3]
sphere diameter	d	[m]

Tab. 1: Relevance list of the flow around a sphere

The first step in the DA-TB is to create a relevance list in Matlab. Therefore the variable names and the corresponding units are given as string cell arrays respectively. Should there be dimensionless variables in the relevance list, these would be assigned a dimension of '1'. The function call `unit2si('help')` returns a list of all implemented dimension names and the respective conversion factors.

Here, the relevance list for the drag force problem is generated, by first specifying the variable names and the corresponding dimensions. Then the dimensions are transformed into basic SI dimensions and the relevance list object is generated.

```
% Define the variable names and the
% respective dimensions
N = {'D', 'v',   'nu',   'rho',   'd'};
u = {'N', 'm/s', 'm2/s', 'kg/m3', 'm'};

% create the relevance list
[d,f] = unit2si(u);
RL = rlist(N,d,f);
```

After specifying the relevance list, the base variables have to be chosen. The number of base variables for the given relevance list can be queried using `numpi(RL)`. In the drag force problem, the three base variables velocity *v*, diameter *d*, and fluid density *r* are chosen and the corresponding dimensional set is calculated.

```
% choose the base variables
bv = {'v','d','rho');

% do the dimensional analysis
piset = diman(RL,bv);
```

The resulting dimensional set is given by

```
'D'    'nu'    'v'    'd'    'rho'

[ 1]   [ 0]   [ 0]   [ 0]   [ 1]
[ 1]   [ 2]   [ 1]   [ 1]   [-3]
[-2]   [-1]   [-1]   [ 0]   [ 0]
[ 1]   [ 0]   [-2]   [-2]   [-1]
[ 0]   [ 1]   [-1]   [-1]   [ 0]
```

The dimensionless groups in this dimensionless set `piset` can now be printed to screen or file using the `pretty` command.

```
% pretty print the pis
pretty(piset);

                D
 pi1 = ---------
         2  2
        v  d  rho

              nu
    pi2 = ---
              v d
```

With this given dimensional set `piset` we can transform the data given in dimensional representation in /8/ to the dimensionless domain.

```
% load demo data
load demodata/spheredata
XData = Kugel';

% tramsform the x-data to SI basic units
XData = data2si(XData,RL);

% and transform the data
PiData = dtrans(XData,piset);
```

The graphical plot of the dimensionless sphere data is shown in figure XXX.

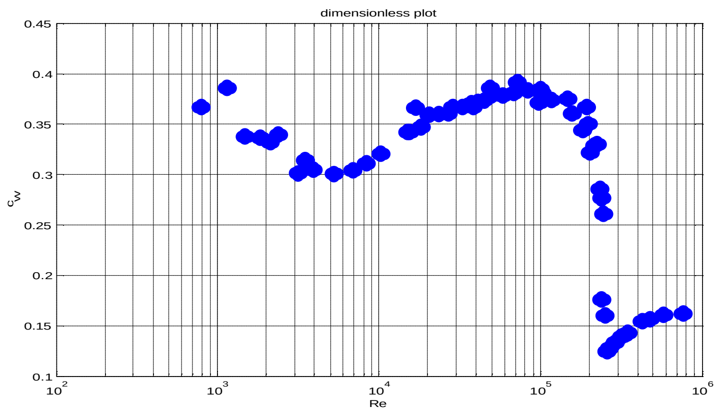

Fig. 3: Transformed data of the sphere drag coefficient vs. Reynolds number

Instead of the command line version shown above, the dimensionless groups ("π"s) can also be found using the `datool` GUI. All relevant variables appear first in the "Variables" column and can then be moved to either the "Base Variables" or the "Dep. Variables" for the dependent variables. Automatically the number of base and dependent variables is calculated and shown. The dimensionless groups for the selected base and dependent variables are shown in the field "Dimensionless Groups".

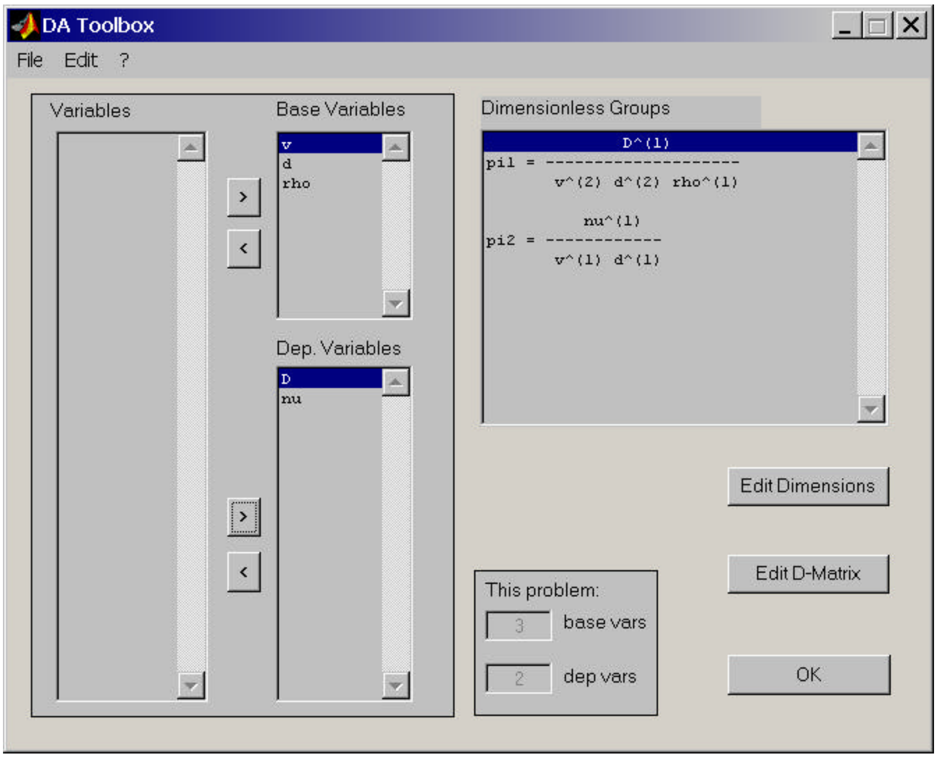

Fig. 4: The `datool` GUI for the DA-TB

The buttons "Edit Dimensions" and "Edit D-Matrix" call an array editor to modify either the dimensions of the variables or the D-matrix respectively. By editing the D-matrix a dimensional set with (at least partly) known dimensional groups (if consistent with the base and dependent variables) can be determined.

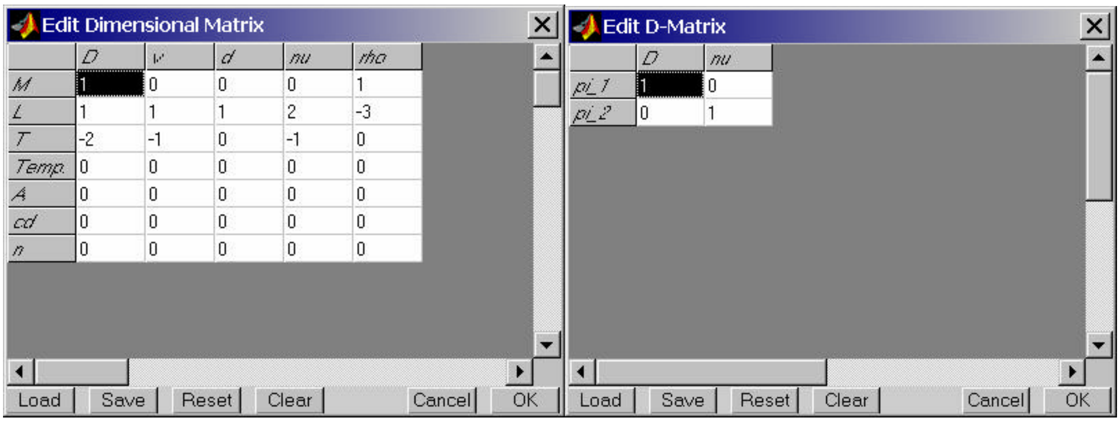

Fig. 5: The build-in array editor

The build-in array editor, shown in fig. 5, allows saving matrices to disk and restoring them again, as well as "Reset" and "Clear" buttons allow undoing the modifications or clearing the matrix.

5. Example: Energy in a nuclear explosion

The following example is taken from Taylor /4,5/ and describes the energy in a nuclear explosion as a function of observable variables, namely the cloud radius, the time since explosion, and the air density.

```
% G.I. Taylor: Energy in a nuclear explosion
%
% define the list of relevant variables and
% their respective dimensions
N = {'R', 't', 'rho',   'E'};
u = {'m', 's', 'kg/m3', 'J'};

% form a relevance list
[d,f] = unit2si(u);
RL = rlist(N,d,f);

% check the number of possible dimensionless groups
numpi(RL)

ans =

     1

%
% and choose the base variables
bv = {'t','rho','E'};
%
% and do the dimensional analysis
piset = diman(RL,bv);

% and print the resulting dimensionless group
pretty(piset);

                          5
                         R  rho
                 pi1 = -------
                           2  .
                          t  E
%
% since we only have one dimensionless group, this
% must be a constant. Using e.g. the symbolic math
% toolbox the relationship
%
%   E = pi1^-1 * rho * R^5 * t^-2
%
% can be established and the constant pi1 can
% be found from one single observation only.
```

From observational data, the energy involved in the explosion can then be estimated, using a mean air density ρ_0 = 1.25 kg / m^3 and setting the factor $\pi_1 \sim 1$. Taking the mean of the estimated energy gives an estimate of 8.28254 x 10^{13} J which corresponds to roughly 19.5 tons of TNT.

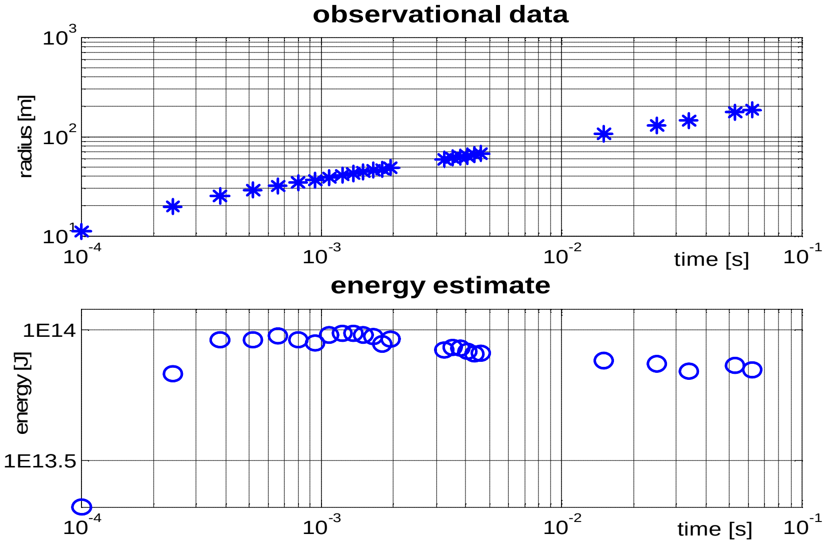

Fig. 6: The measured blast wave radius (above) and the estimated blast energy (below).

This example is included with the toolbox as `blastdemo`.

6. Availability, requirements, and known limitations

The Dimensional Analysis Toolbox for Matlab is available at the primary web site
http://www.sbrs.net/

Alternatively, the toolbox is also available from MatlabCentral at
http://www.mathworks.com/matlabcentral/

The DA-TB has been implemented to minimize requirements. Only a basic Matlab system (or possibly even the free Octave/6/ system) is needed to access the basic functionality of the DA-TB. The graphical UI of the DA-TB requires Matlab R12 or newer, the Symbolic Toolbox is needed for a better formatted pretty print and LaTeX output, and a Windows operating system is needed to access the matrix editor functionality of the GUI.

Another requirement is at least basic knowledge of the user about the method of dimensional analysis.

The unit information the `unit2si` function file is far from being complete. Currently 136 units are implemented; any extensions to that list are welcome. `unit2si` does not parse the unit names for the use prefixes (like µ for 1e-6) yet.

7. Summary

The Dimensional Analysis Toolbox for Matlab helps the researcher or engineer to perform basic dimensional analysis based on the knowledge of the relevant variables and their respective dimensions, as well as numeric data transformation from the dimensional to the dimensionless domain in the same tool that is then used for an in-depth data analysis and modeling. The DA-TB features an easy-to-use graphical user interface and a formatted output for the symbolic representation of the dimensionless groups.

Literature

/1/ Brückner, S. "The Dimensional Analysis Toolbox for Matlab", User's Manual, Stuttgart, 2002, http://www.sbrs.net/

/2/ MATLAB R13 (6.5), The Mathworks Inc., Natick, MA, USA, 2002, http://www.mathworks.com/

/3/ Buckingham, E., "On Physically Similar Systems: Illustration of the Use of Dimensional Equations", Phys. Review 4, 345-376, 1914.

/4/ Taylor, G., "The formation of a blast wave by very intense explosion, I. Theoretical dicussion", Proc. Roy. Soc. **201**, 159-174, 1950.

/5/ Taylor, G., "The formation of a blast wave by very intense explosion, II. The atomic explosion of 1945", Proc. Roy. Soc. **201**, 175-186, 1950.

/6/ Eaton, J.W., "GNU Octave Manual", Network Theory, London, 2002, http://www.octave.org/

/7/ Szirtes, T., "Applied Dimensional Analysis and Modeling", MacGraw-Hill, New York, 1997.

/8/ Prandtl, L., Wieselsberger, C., Betz, A., "Ergebnisse der Aerodynamischen Versuchsanstalt zu Göttingen", Oldenbourg, Berlin, 1923.