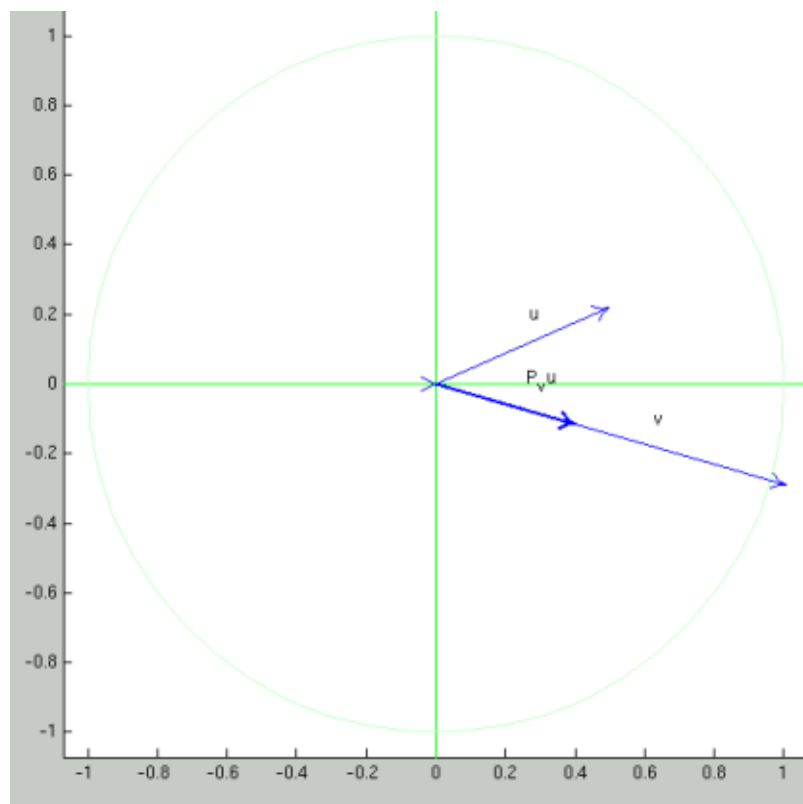


# VECTORS AND MATRICES

## COMPUTER SESSION C1



### BACKGROUND

### PREPARATIONS

The session is divided into two parts. The first part involves experimenting in the Mathematics Laboratory and the second part involves programming with vectors and matrices.

Start Matlab.

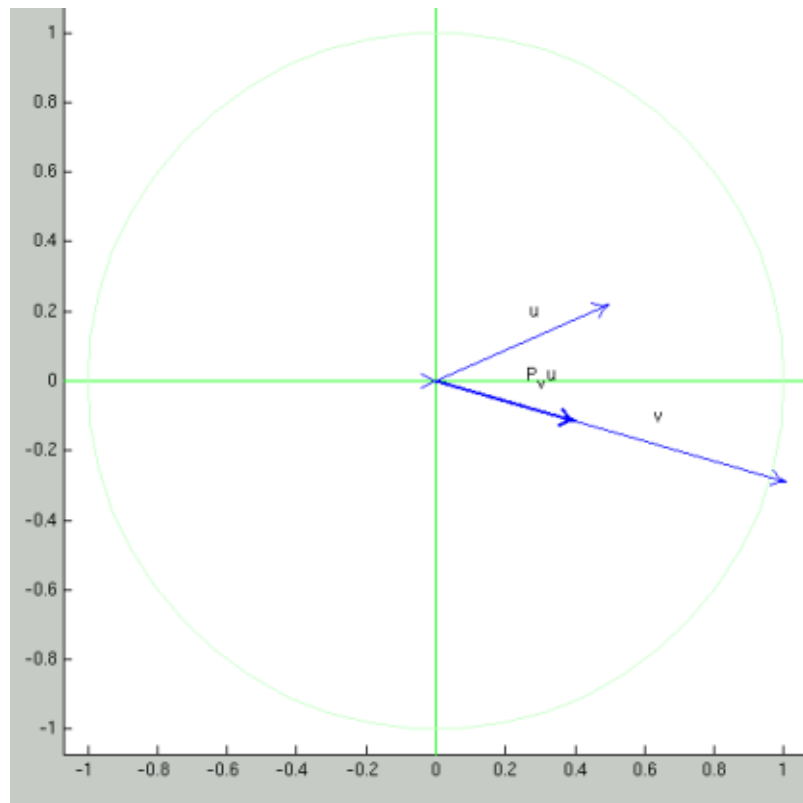
If you are working on the computers of the School of Chemical Engineering at Chalmers, then download the file `startmath.m` to your Matlab work directory (if you have not done this already). This file is available on the web page of this session under Programs and templates. Then type `startmath` at the Matlab prompt. This command sets the search path to the directories where the Mathematics Laboratory is kept.

If you are working on another computer, then download the file `MathematicsLaboratory.zip` to your Matlab work directory. This file is available on the web page of this session under Programs and templates. Unzip the file, it should create a directory `guis` in your Matlab work directory. At the Matlab prompt, type: `addpath guis`. You are now ready to use the Mathematics Laboratory.

Keep your AMBS book with you and open at the relevant chapters.

## PROBLEMS

### **Problem 1 - Vectors and matrices in the Mathematics Laboratory.**



- (1) Give the command `open('RM+.fig')` to open the Road Map to the Mathematics Laboratory, and press the Vector C button to activate the Vector Calculator. Alternatively you may start the Vector Calculator directly from the matlab prompt by the command `open('V.fig')`.
- (2) Start by pressing clear. Create a new vector by giving a return in the new `v = edit` box. Press the edit button and change the vector by point-dragging the arrow of the vector. Notice the component representation of the vector and make sure you fully understand how it works.
- (3) Create another vector but this time by pressing the create button and then defining the vector simply by drawing it with the mouse with the (left) mouse button down, or just clicking at the desired point of the vector arrow. If you would like to define another vector, just release the mouse button and draw again.
- (4) To change to edit mode, press the edit button and you can edit any vector you like. The select button is for just selecting/highlighting a vector without wanting to change it. Also, try the delete function.

- (5) Next try the scaling function by giving a scale factor in the scale by edit box (followed by a return), which scales the current vector by the given scale factor. The invert function is a special case corresponding to scaling by -1. Also the normalize function is a special case corresponding to scaling by  $\frac{1}{\|v\|}$ , where  $\|v\|$  is the norm, or length of  $v$ . Try to normalize a given vector ( [ .5 .5 ] for example) manually through the scale by function!
- (6) Next see to it that you have two vectors, one with the name  $v$  and another one by the name  $u$ . You name a vector by entering the desired name in the save as edit box (and giving a return). The default name at creation time is  $v$ , as you may have noticed. Now add  $u$  and  $v$  by first highlighting  $u$  (by the select function), then pressing the add button, and then clicking at the  $v$  vector. Note how the sum  $u + v$  of  $u$  and  $v$  is defined. To further explore how  $u + v$  depends on  $u$  and  $v$ , press the edit button and move around  $u$  and  $v$  by point-dragging. Repeat with the (minus) operation.
- (7) Next study the projection of  $u$  onto (the direction of)  $v$  by following the same scheme; first select  $u$ , then press the  $\rightarrow$  proj  $\rightarrow$  button, and then click at  $v$ . Again, study how the projection depends on  $u$  and  $v$  by pressing the edit button and then moving around  $u$  and  $v$ . Which three factors determine the length of the projection vector? What is its direction? When does the projection vector degenerate to length 0?
- (8) To study the transformation of a vector when multiplied by a matrix, press the  $Au \rightarrow u$  button. Seek matrices that a) do not change  $u$  at all, b) rotate  $u$  half a turn (invert  $u$ ), c) rotate  $u$  90 degrees in the counter-clockwise direction, d) rotate  $u$  45 degrees in the clockwise direction, e) project  $u$  onto the  $x$  - axis.
- What does the matrix  $[\cos(.1) \ -\sin(.1); \ \sin(.1) \ \cos(.1)]$  do to a vector  $u$ ?
- What does the function  $\text{inv}(A) \ u \rightarrow u$  transformation do?
- Are there matrices that leave some vectors  $u$  unchanged but change others? Hint: try  $A = [ \ 1 \ 0; \ 0 \ -1 ]$ .
- For the matrix  $A = [ \ 1 \ 0; \ 0 \ -.9 ]$ , find vectors which do not change direction while transformed by  $A$ , and vectors that flip to the opposite direction when transformed by  $A$ .
- (9) Now turn to complex multiplication. Define vectors  $u$  and  $v$ . Highlight  $u$  (by the select button and clicking at  $u$  ), then press the  $\text{compl}^*$  button and then click at  $v$ . Then study how  $u * v$  depends on  $u$  and  $v$  by pressing the edit button and moving  $u$  and  $v$  around.

Make sure you get to understand what you see! Then repeat with the `comp1/` operation.

- (10) Now answer the following questions: a) Which vector should be added to  $[1 \ 1]$  to give  $[0 \ .5]$ . Draw your answer and name it `MyAnswer`. To check your answer, define vectors  $u = [0 \ .5]$  and  $v = [1 \ 1]$  (using `new=` and save as) and then compute  $u-v$ . b) What is the projection of  $u = [-1 \ .5]$  onto  $v = [1 \ 0]$ ,  $[0 \ 1]$  and  $[1 \ 1]$ , respectively. Again, check your answer by defining  $u$  and  $v$  as stated, and computing the projection of  $u$  onto  $v$ . c) Which complex number should be multiplied by  $(1-i)^+$  to give  $-1$ . Check your answer by defining  $u = [-1 \ 0]$  and  $v = [1 \ 1]$  (corresponding to  $-1$  and  $1+i$ , respectively), and then computing  $\frac{u}{v}$ . d) How does the matrix  $[1 \ -1; \ -2 \ 0]$  transform the vector  $[.5 \ .5]$ ? Check your answer by defining corresponding  $A$ 's and  $u$ 's and pressing `Au - > u`.

### Problem 2 - Vector and matrix syntax and functions.

- (1) To create a list-variable/vector:

$$v = [ 2 \ -3 \ 7 ]$$

in Matlab you may specify each entry/component of  $v$  separately by writing (at the matlab prompt) `v(1) = 2`, then `v(2) = -3`, then `v(3) = 7`, or you may use the `[ ]` syntax, like in `w = [1.5 3.9 -6.1]`, with the components separed by blanks or commas. Try both the `v(index) = value` and the `w = [ . . . ]` method!

- (2) Recall that `length(v)` gives the number of components (the “dimension”) of the vector  $v$ . Try it! Try `help length` to see documentation for the `length(v)` function.
- (3) Recall that you may redefine individual components of an existing vector like  $w$  by directly addressing the component you would like to change, for example by writing `w(2) = 5.7`. Try this.
- (4) Vectors may be scaled by multiplication with a common scale factor, that is, a real number. For example you may double all the components of  $v$  by writing `2 * v`. Try this and check the result. Note that if you write `2 * v` the result is put in the “answer box” `ans`, while if you actually want to change  $v$ , that is, to have  $v$  hold the result, you would write `v = 2 * v`. How could you get back the old  $v$  after this? Try!

- (5) Further recall that vectors (of the same length) can be added by writing for example  $v + w$ . Try this and check the result! Also try  $v - w$ . Does this perform the action you expected? Good! Of course you may also combine scaling and adding into so called linear combinations of  $v$  and  $w$  by writing, for example,  $2 * v + 3 * w$ . Try this also.
- (6) The scalar product of two vectors  $v$  and  $w$  can be obtained by writing `sum(v .* w)` or `dot(v, w)`. Recall that `v .* w` computes the vector of products  $[v(1) * w(1), v(2) * w(2), v(3) * w(3)]$ , and that `sum` then adds the products of this vector together. Before you check all this you may want to redefine  $v$  and  $w$  for example as  $v = [1 \ 2 \ 3]$  and  $w = [4 \ 5 \ 6]$  so that you can more easily check what is going on. Do you then get the scalar product of these to be  $4 + 10 + 18 = 32$  as expected? Try both `sum(v .* w)` and `dot(v, w)`! Is `dot(w, v)` the same as `dot(v, w)`? Always?
- (7) Can you now compute the projection  $P_v$  of  $v$  in the direction of  $w$ ? Recall that the projection should go in the direction of  $w$  so that  $P_v = cw$  for some  $c$ , and that the formula for  $c$  is  $c = \frac{v \cdot w}{w \cdot w}$  (`dot(v, w) / dot(w, w)` in Matlab syntax), or if you like,  $c = \frac{v \cdot w}{\|w\|^2}$  (`dot(v, w) / norm(w)^2` in Matlab syntax). Recall that  $\|w\|$  is the Euclidean length of  $w$  given by  $\sqrt{w \cdot w}$  (`sqrt(dot(w, w))` in Matlab syntax). So, what is the projection of  $v$  in the direction of  $w$ ?
- (8) The vector product, or cross product of two (3-element)vectors  $v$  and  $w$  is given by `cross(v, w)` in matlab. Try it, and check that the result is  $[-3, \ ? \ ?]$  as expected. What should the question marks be here?
- (9) Recall that vectors may also appear as columns. For example, you may view the column counterpart of  $w$  by writing  $w'$  in Matlab, and you may transform  $w$  to column form by writing  $w = w'$ . Try this! Similarly, you may transform a column vector back to a row vector by another `'`. That is,  $w = w'$  brings  $w$  back to its “row form”. Note that also column vectors may be scaled and added (if they have the same dimension) as before. However, matlab refuses to add row vectors to column vectors, and vice versa. Vector algebra requires vectors of the same form or type!
- (10) Let us now consider a “column vector of row vectors”, or a “row of column vectors”. Such a construction is called a matrix. Rather, a matrix is a rectangular scheme of numbers organized in  $m$  rows and  $n$  columns, but which therefore can be considered as a row of

$n$  columns, each of height  $m$ , or alternatively, a column of  $n$  row vectors, each of length  $n$ . Check that this makes sense to you!

- (11) You may create a matrix  $A$  with 2 rows and 3 columns by giving each of the 6 entries like  $A(1, 1) = 11$ ,  $A(1, 2) = 12$ ,  $A(1, 3) = 13$ ,  $A(2, 1) = 21$ , etc. Note that the first index  $i$  in  $A(i, j)$  is the row number and the second index  $j$  is the column index. That is  $A(2, 1) = 21$  specifies that the number in column 1 and row 2 of the matrix should be 21. You may also specify the matrix using the `[ ]` syntax as  $A = [11\ 12\ 13; 21\ 22\ 23]$ , giving the entries row by row with a semi-colon `;` indicating the start of a new row. Try this!
- (12) Like vectors (which are special cases of matrices with only one row or only one column), matrices can be scaled, by multiplication of a (real) number/scale factor, and added to other matrices (of the same form, i.e. with the same number of rows and columns). A matrix with  $m$  rows and  $n$  columns is called an  $m \times n$  matrix, expressed as “ $m$ -by- $n$  matrix”.
- (13) Let us now consider the important “matrix-vector multiplication”. The product  $A * v$ , where  $A$  is an  $m \times n$  matrix and  $v$  is an  $n \times 1$  column vector (that is, a matrix with only one column), produces an  $m \times 1$  column vector. That is, the matrix  $A$  transforms  $n \times 1$  column vectors to  $m \times 1$  column vectors. What happens here could be viewed (at least) two ways. One way to see what happens is that  $A * v$  produces a linear combination of the columns of  $A$  with the coefficients given by  $v$ .

For example, for  $A = [1\ 2\ 3; 4\ 5\ 6]$  with (recall the `'`) columns  $[1\ 4]'$ ,  $[2\ 5]'$  and  $[3\ 6]'$ , and  $v = [1\ 2\ 3]'$ ,  $A * v$  gives the sum of  $1 * [1\ 4]'$ ,  $2 * [2\ 5]'$  and  $3 * [3\ 6]'$ , which is  $[1\ 4]'$   $[4\ 10]'$  +  $[9\ 18]'$  =  $[14\ 32]'$ . Check this!

The other way to view what happens as a result of writing  $A * v$  is that each row of  $A$  is multiplied with  $v$  using the scalar product multiplication, and that the  $m$  scalar products produced this way (recall that  $A$  has  $m$  rows), is delivered in a  $m \times 1$  column vector. Check that this is indeed the case!

- (14) Seek (by some trial and error) a linear combination of the columns of  $A$  that equals  $b = [10\ 20]'$ . That is, successively change  $v$  until  $A * v$  equals  $b$ . In fact, there are many such combinations to find. This is because there are 3 columns to use and only 2 numbers to equal (10 and 20).
- (15) If, on the other hand,  $A$  has only 2 columns, there is, in general, a unique combination of the two columns that gives the desired result  $b = [10\ 20]'$ . Matlab has a built-in functionality for finding

this combination  $v$ , using the Matlab “backslash” To get  $v$  you write  $v = A \setminus b$ . Note that  $v$  has now only two entries because there are only two columns in  $A$  to combine!

## SOLUTIONS

Make sure that you really try to solve each problem before looking at the solutions. Have you really tried to solve the problem or should you try again before looking at the solution?

The solutions are available on the web page of this session under *[Solutions to problems](#)*.

## ABOUT

This Computer Session is part of the Body and Soul educational program. More information can be found at

<http://www.phi.chalmers.se/body soul/>

This Computer Session is maintained by Johan Jansson (johanjan@math.chalmers.se).