1

For each solved variable, we distinguish between the number of "solutions" and the number of "versions" it has. The number of solutions is determined by the mathematical form of the solution for example:

$$x = \sin(\theta) \tag{1}$$

has two versions arising from its two solutions

$$\theta = [\arctan(x), \pi - \arctan(x)] \tag{2}$$

However a variable whose solution is single valued, could have multiple versions of it depends on a variable with multiple versions. For example,

$$y = \theta + \pi \tag{3}$$

y has one solution, but two versions

$$y = [\pi + \arctan(x), 2\pi - \arctan(x)] \tag{4}$$

depending on which version of θ is used.

Furthermore, a solution may depend on more than one solved variable and all the combinations of versions of those variables generate versions of the current variable. To summarize, if a variable has n_s solutions and n_d versions due to its dependencies, Then it has $n_s \times n_d$ versions. If a variable x_i has m different variables as dependencies, then it as

$$n_{vi} = n_{sj}\Pi_{j=1}^{m} n_{vj} \tag{5}$$

We illustrate this with a small system of equations which is easy to solve but has these versioning characteristics of IK problems. Let the problem be to find all sets of the unknowns $[x_1 \ldots x_5]$ which satisfy the following five equations.

$$x_1 - x_4 - x_2 = 0 \tag{6}$$

$$x_2 - x_4/2 = 0 \tag{7}$$

$$9 - x_3 = 0 \tag{8}$$

$$x_4 - \sqrt{x_3} = 0 \tag{9}$$

$$\sqrt{4} - x_5 = 0 \tag{10}$$

where each of the x_i are unknowns and the other terms (here they are numbers) are known. By inspection, we can solve these in the order $[(x_3, x_5)(tie), x_4, x_2, x_1]$.

1. x_3 is a trivial solution: $x_3 = 9$ and there is only one solution.
2. x_5 is simple, but there are two solutions: $[\sqrt{4}, -\sqrt{4}]$.
3. x_4 similarly has two solutions: $[\sqrt{9}, -\sqrt{9}]$.
4. x_2 has only one solution, $x_4/2$, but we have two *versions* due to the two solutions of x_4: $[1.5, -1.5]$
5. x_1 depends on both x_4 and x_2 so though it has one solution, it has four versions (in general distinct): $[4.5, -4.5, -4.5, 4.5]$ due to the combinations of versions of its two dependencies.

var.,	multiplicity	Solutions or Versions
x_3	1	$[9]$
x_5	2	$[2, -2]^S$
x_4	2	$[3, -3]^S$
x_2	1	$[1.5, -1.5]^V$
x_1	1	$[4.5, -4.5, -4.5, 4.5]^V$

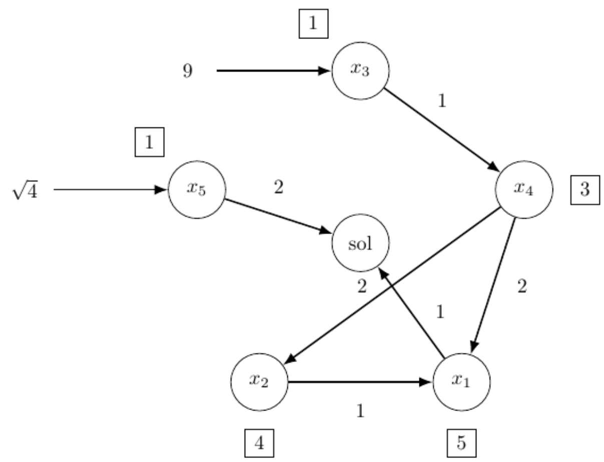

Figure 1: Dependencies to solution of Equations 6 to 10. Edge weights are n_{si} of the origin of the edge. The square annotation gives the solution order of each variable. "sol" node is the full solution set.

The solutions dependencies can be represented as a graph (Figure 1).

Next, we assemble solution vectors by following the solution order as follows:

1. x_3 has one solution, $x_{3s1} = 9$ giving one version

$$x_{3v1} = x_{3s1} = 9 \tag{11}$$

   We will continue by collecting these versions into a set of vectors:

$$[x_{3v1}] \tag{12}$$

2. x_5 has two solution but depends on no previous unknowns. Its versions are thus $x_{5v1} = x_{5s1} = 2, x_{5v2} = x_{5s2} = -2$. We thus double the number of rows giving:

$$\begin{bmatrix} x_{3v1} & x_{5v1} \\ x_{3v1} & x_{5v2} \end{bmatrix} \tag{13}$$

   note that the order of these two variable columns does't matter since they are independent of each other.

3. x_4 has two solutions based on the multiple solutions of Eqn 9 which depend on x_3 (having only one version) but these solutions are independent of x_4 and x_5. We thus have to double the rows again :

$$\begin{bmatrix} x_{3v1} & x_{5v1} & x_{4v1} \\ x_{3v1} & x_{5v2} & x_{4v2} \\ x_{3v1} & x_{5v1} & x_{4v1} \\ x_{3v1} & x_{5v2} & x_{4v2} \end{bmatrix} \tag{14}$$

4. x_2, similarly has one solution but depends on the versions of x_4

$$\begin{bmatrix} x_{3v1} & x_{5v1} & x_{4v1} & x_{2v1} \\ x_{3v1} & x_{5v2} & x_{4v2} & x_{2v2} \\ x_{3v1} & x_{5v1} & x_{4v1} & x_{2v1} \\ x_{3v1} & x_{5v2} & x_{4v2} & x_{2v2} \end{bmatrix} \tag{15}$$

5. x_1 has only one solution, but it depends on two variables: x_2 which has 2 versions, and x_4 which has two versions. Applying Eqn 5. we have 4 versions:

$$\begin{aligned} x_{1v1} &= x_{4v1} + x_{2v1} \\ x_{1v2} &= x_{4v2} + x_{2v2} \\ x_{1v3} &= x_{4v1} + x_{2v2} \\ x_{1v4} &= x_{4v2} + x_{2v1} \end{aligned} \tag{16}$$

but we do not change the number of rows because of the single solution to Eqn 6.

$$\begin{bmatrix} x_{3v1} & x_{5v1} & x_{4v1} & x_{2v1} & x_{1v1} \\ x_{3v1} & x_{5v2} & x_{4v2} & x_{2v2} & x_{1v2} \\ x_{3v1} & x_{5v1} & x_{4v1} & x_{2v1} & x_{1v3} \\ x_{3v1} & x_{5v2} & x_{4v2} & x_{2v2} & x_{1v4} \end{bmatrix} \tag{17}$$

The above process can be summarized as:

Once an algorithm such as IKBT has found solutions to all unknowns we can then collect valid solution vectors as follows:
For each variable x_i in solution order:

1. Determine its number of solutions from its solution method. Example: $\sqrt{x}$ has 2 solutions)
2. Determine its number of dependencies on previously solved unknowns and determine the number of versions of each dependency: n_{vj}. Example: $y_5(y_2, y_4)$ where y_2 has 4 versions and y_4 has 2 versions.
3. Compute the number of versions for x_i, n_{vi}, using Eqn 5.
4. Number the versions x_{ivj} where i selects the variable and j selects the version number.
5. Enumerate and save the expression for each version. Example:
   - $x_{4v1} = -\sqrt{9}$
   - $x_{4v2} = \sqrt{9}]$
6. Add them as a new colum to the solution vector matrix, if n_{vi} is less than the number of rows, repeat the versions to complete all rows. If n_{vi} is greater than the number of rows of the solution vector matrix, copy the rows n_{si} times and append them to to get n_{vj} rows.

2 Solution Graphs

A graph of the solution dependncies can be constructed. For the system of equations above, and the solution list, we have