Appendix A

A Computer Program for Determining the Planar Stewart Platform Workspace (*PLANSTEW*)

A.1 INTRODUCTION

This appendix explains the automated computer program *PLANSTEW* that was used to map the accessible output sets as well as the bifurcation point connecting curves of the planar Stewart platform.

PLANSTEW consists of a *main program* and a few *subroutines*. The outlay of the main program is shown in Figure A.1, and the different subroutines are shown in Figure A.2, Figure A.3, Figure A.6 and Figure A.7. The detail of the main program is explained here and the discussions of the subroutines are included as sub-paragraphs.

A.2 THE MAIN PROGRAM

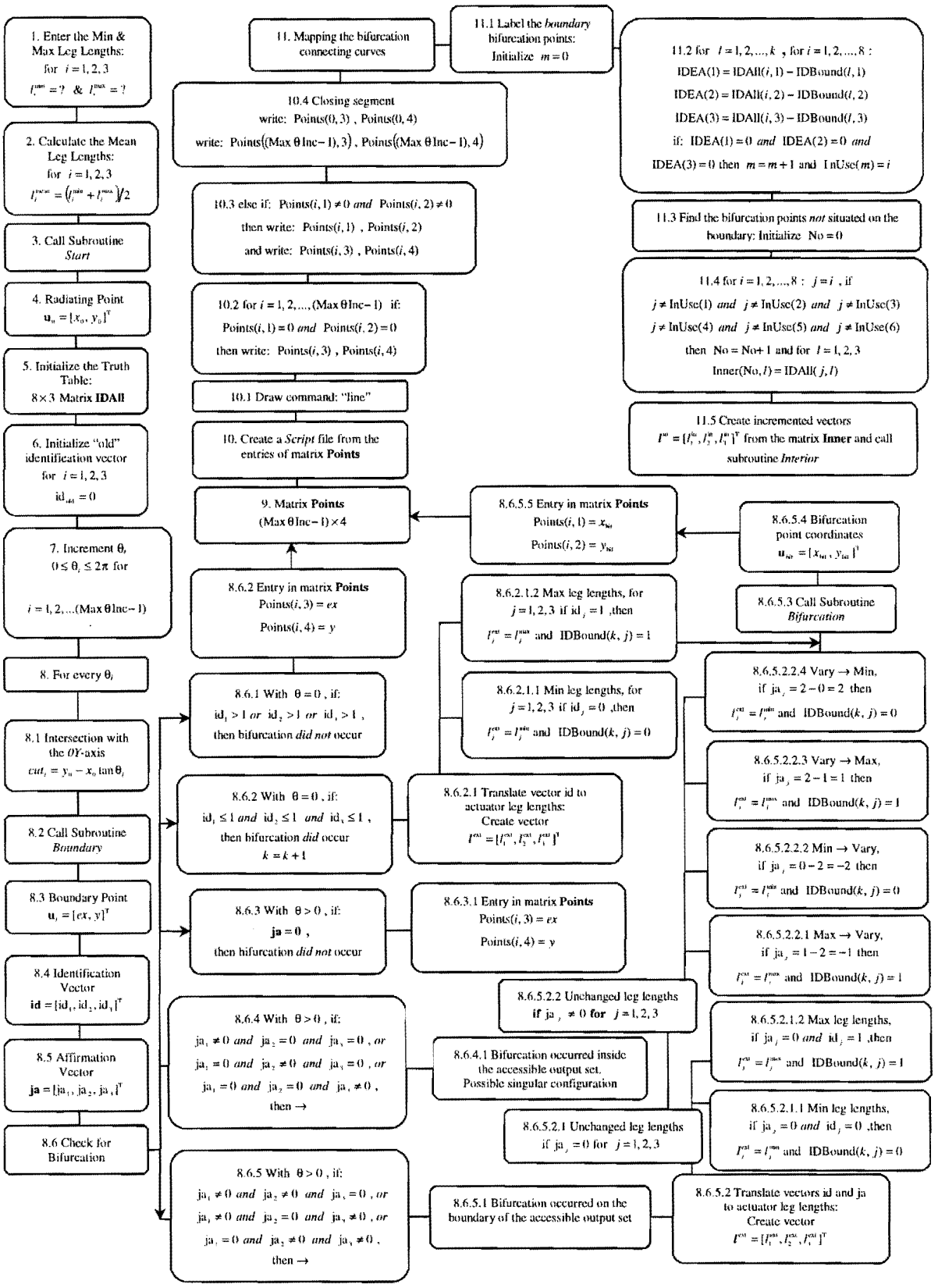

Figure A.1 Flow chart showing the layout of the main program.

Looking at Figure A.1, the first thing the user has to enter, is the respective minimum and maximum actuator leg lengths. The main program then calculates the mean actuator leg lengths.

$$l_i^{\text{mean}} = \frac{l_i^{\min} + l_i^{\max}}{2} \tag{A.1}$$

for $i = 1, 2, 3$

Equation (2.1) is used in equation (2.10), remembering that the actuator leg lengths were chosen as the input variables. Subroutine *Start* is used to determine the initial central point $\mathbf{u}^0$.

A.2.1 Subroutine *Start*

In the flow chart showing the lay out of subroutine *Start* (see Figure A.2), it is evident that the user has to enter an initial guess as to where the central point $\mathbf{u}^0$ is situated. This initial guess preferably has to be inside the accessible output set, and for the planar Stewart platform under consideration, the initial guess that was entered, is $(x, y) = (1.0,\ 1.2)$.

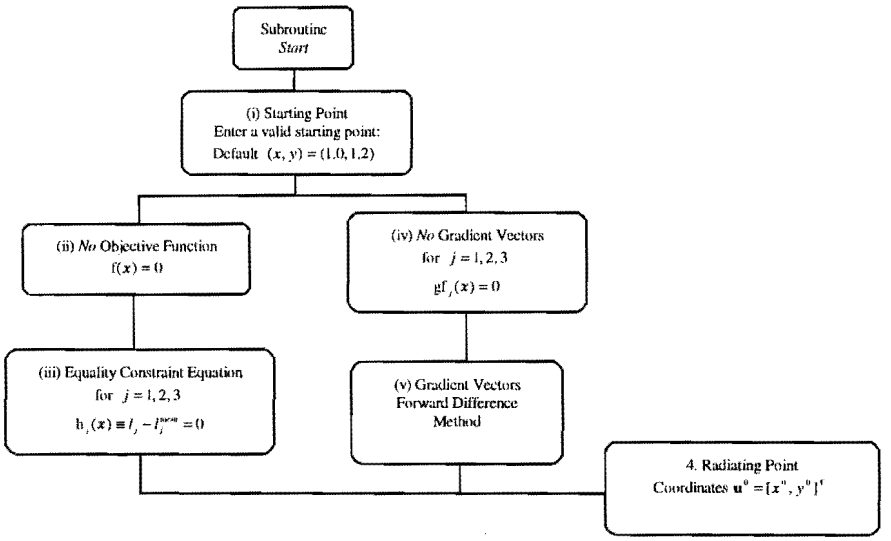

Figure A.2 Subroutine *Start*.

There is no explicit objective function, as the three non-linear equations are entered as equality constraints, i.e.

$$\begin{aligned} v_1(\mathbf{u}, w) - v_1^{\text{mean}} &= 0 \\ v_2(\mathbf{u}, w) - v_2^{\text{mean}} &= 0 \\ v_3(\mathbf{u}, w) - v_3^{\text{mean}} &= 0 \end{aligned} \tag{A.2}$$

The gradient vectors of the equality constraints are determined numerically using the forward difference method:

$$\frac{\partial f(\boldsymbol{x}_i)}{\partial x_j} \approx \frac{\left(f(\boldsymbol{x}_i + \Delta \boldsymbol{x}_j)\right) - f(\boldsymbol{x}_i)}{\Delta x_j} \tag{A.3}$$

The three non-linear equations are solved by minimizing the square of the Euclidean norm (2.11), and the output of subroutine *Start* is the radiating point $\mathbf{u}^0$ from where the boundary of the planar Stewart platform is mapped.

The next step in the main program (step 5 in Figure A.1), is to initialize the "truth table". Knowing that the planar Stewart platform has three legs each having two extreme positions, it is evident that there are $2^3 = 8$ bifurcation points. This truth table is used to identify which bifurcation points are situated on the boundary of the accessible output set. The remaining bifurcation points are used to trace the bifurcation connection curves as will be explained later.

The initialization of the truth table involves creating an 8×3 matrix **IDAll** where each row represents a different bifurcation point. The entry in each of the three columns indicates whether the corresponding leg takes on a minimum or maximum length with the manipulator working point corresponding with that specific bifurcation point. Based on the proposed labeling notation (Section 2.5.3.1), a 1 entry in column i indicates that leg i takes on a maximum length, and a 0 entry that leg i takes on a minimum length.

Step 6 in Figure A.1 is the initialization of the vector $\mathbf{id}_{\text{old}}$. This is an "old" identification vector, and it is used in subroutine *Boundary*. An auxiliary variable θ is defined in the main program to be used in the mapping of the planar accessible output set as discussed in Section (2.4). This orientation angle θ is incremented from 0 to 2π, as follows:

$$\theta_i = \frac{i(2\pi)}{\text{Max}\,\theta\,\text{Inc}} \tag{A.4}$$

for $i = 0, 1, 2, \ldots, (\text{Max}\,\theta\,\text{Inc} - 1)$.

The user decides on the number of increments required, and defines the parameter "MaxθInc". It follows that once the parameter "MaxθInc" is specified, the increment size δ of the emanating rays used for mapping the planar accessible output set (see Section 2.4), is fixed.

$$\delta = \frac{2\pi}{\text{Max}\,\theta\,\text{Inc}} \tag{A.5}$$

For each orientation angle θ_i, the intersection of the specific emanating ray with the *0Y*-axis (Cut_i) is determined in the main program (7.1 in Figure A.1), as it is used in subroutine *Boundary*.

$$Cut_i = y^0 - x^0 \tan(\theta_i) \tag{A.6}$$

for $i = 0, 1, 2, \ldots, (\text{Max}\,\theta\,\text{Inc} - 1)$

A.2.2 Subroutine *Boundary*

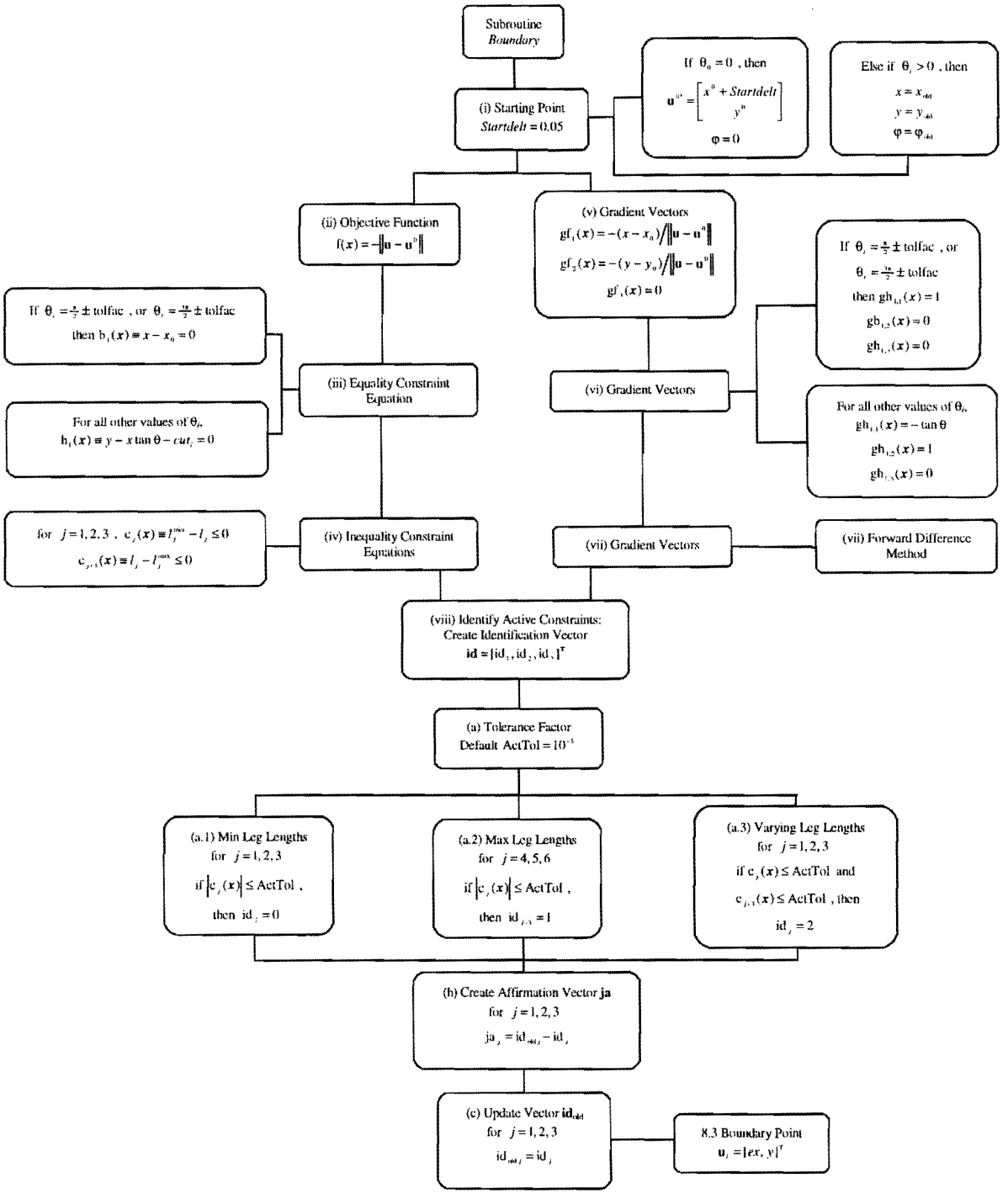

Figure A.3 Subroutine *Boundary*.

With $\theta_0 = 0$, an initial point on ∂A is sought, and optimization *problem (i)* as described in Section 2.3 is applicable. In order to make sure that the maximization is not done π out of phase, an offset (*Startdelt*) is added to the x-value of the radiating point $\mathbf{u}^0$. The actual radiating point used to find the initial point on ∂A with $\theta_0 = 0$, is $\mathbf{u}^{0*}$:

Looking at Figure A.1, the first thing the user has to enter, is the respective minimum and maximum actuator leg lengths. The main program then calculates the mean actuator leg lengths.

$$l_i^{\text{mean}} = \frac{l_i^{\min} + l_i^{\max}}{2} \tag{A.1}$$

for $i = 1, 2, 3$

Equation (2.1) is used in equation (2.10), remembering that the actuator leg lengths were chosen as the input variables. Subroutine *Start* is used to determine the initial central point $\mathbf{u}^0$.

A.2.1 Subroutine *Start*

In the flow chart showing the lay out of subroutine *Start* (see Figure A.2), it is evident that the user has to enter an initial guess as to where the central point $\mathbf{u}^0$ is situated. This initial guess preferably has to be inside the accessible output set, and for the planar Stewart platform under consideration, the initial guess that was entered, is $(x, y) = (1.0,\ 1.2)$.

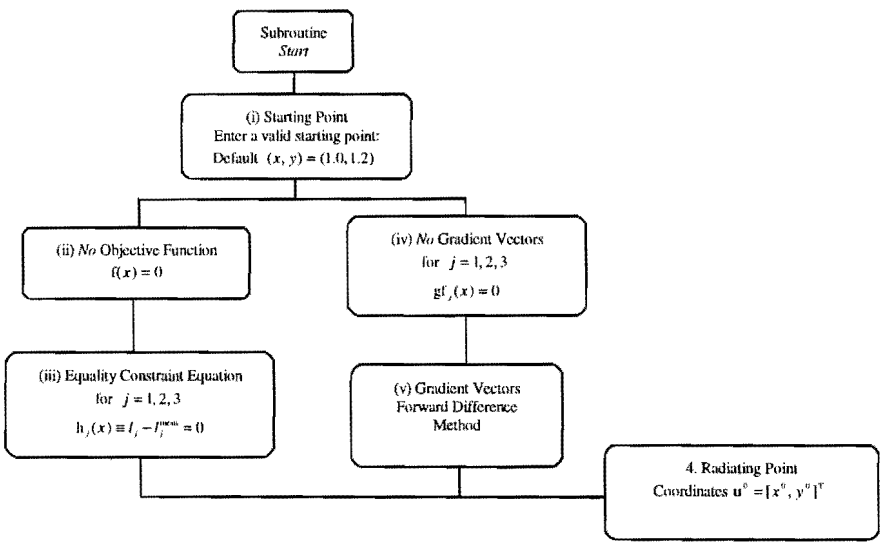

Figure A.2 Subroutine *Start*.

There is no explicit objective function, as the three non-linear equations are entered as equality constraints, i.e.

$$\begin{aligned} v_1(\mathbf{u}, w) - v_1^{\text{mean}} &= 0 \\ v_2(\mathbf{u}, w) - v_2^{\text{mean}} &= 0 \\ v_3(\mathbf{u}, w) - v_3^{\text{mean}} &= 0 \end{aligned} \tag{A.2}$$

The gradient vectors of the equality constraints are determined numerically using the forward difference method:

$$\frac{\partial f(\boldsymbol{x}_i)}{\partial x_j} \approx \frac{\left(f(\boldsymbol{x}_i + \Delta \boldsymbol{x}_j)\right) - f(\boldsymbol{x}_i)}{\Delta x_j} \tag{A.3}$$

The three non-linear equations are solved by minimizing the square of the Euclidean norm (2.11), and the output of subroutine *Start* is the radiating point $\mathbf{u}^0$ from where the boundary of the planar Stewart platform is mapped.

The next step in the main program (step 5 in Figure A.1), is to initialize the "truth table". Knowing that the planar Stewart platform has three legs each having two extreme positions, it is evident that there are $2^3 = 8$ bifurcation points. This truth table is used to identify which bifurcation points are situated on the boundary of the accessible output set. The remaining bifurcation points are used to trace the bifurcation connection curves as will be explained later.

The initialization of the truth table involves creating an 8×3 matrix **IDAll** where each row represents a different bifurcation point. The entry in each of the three columns indicates whether the corresponding leg takes on a minimum or maximum length with the manipulator working point corresponding with that specific bifurcation point. Based on the proposed labeling notation (Section 2.5.3.1), a 1 entry in column i indicates that leg i takes on a maximum length, and a 0 entry that leg i takes on a minimum length.

Step 6 in Figure A.1 is the initialization of the vector $\mathbf{id}_{\text{old}}$. This is an "old" identification vector, and it is used in subroutine *Boundary*. An auxiliary variable θ is defined in the main program to be used in the mapping of the planar accessible output set as discussed in Section (2.4). This orientation angle θ is incremented from 0 to 2π, as follows:

$$\theta_i = \frac{i(2\pi)}{\text{Max}\,\theta\,\text{Inc}} \tag{A.4}$$

for $i = 0, 1, 2, \ldots, (\text{Max}\,\theta\,\text{Inc} - 1)$.

The user decides on the number of increments required, and defines the parameter "MaxθInc". It follows that once the parameter "MaxθInc" is specified, the increment size δ of the emanating rays used for mapping the planar accessible output set (see Section 2.4), is fixed.

$$\delta = \frac{2\pi}{\text{Max}\,\theta\,\text{Inc}} \tag{A.5}$$

For each orientation angle θ_i, the intersection of the specific emanating ray with the *0Y*-axis (Cut_i) is determined in the main program (7.1 in Figure A.1), as it is used in subroutine *Boundary*.

$$Cut_i = y^0 - x^0 \tan(\theta_i) \tag{A.6}$$

for $i = 0, 1, 2, \ldots, (\text{Max}\,\theta\,\text{Inc} - 1)$

A.2.2 Subroutine *Boundary*

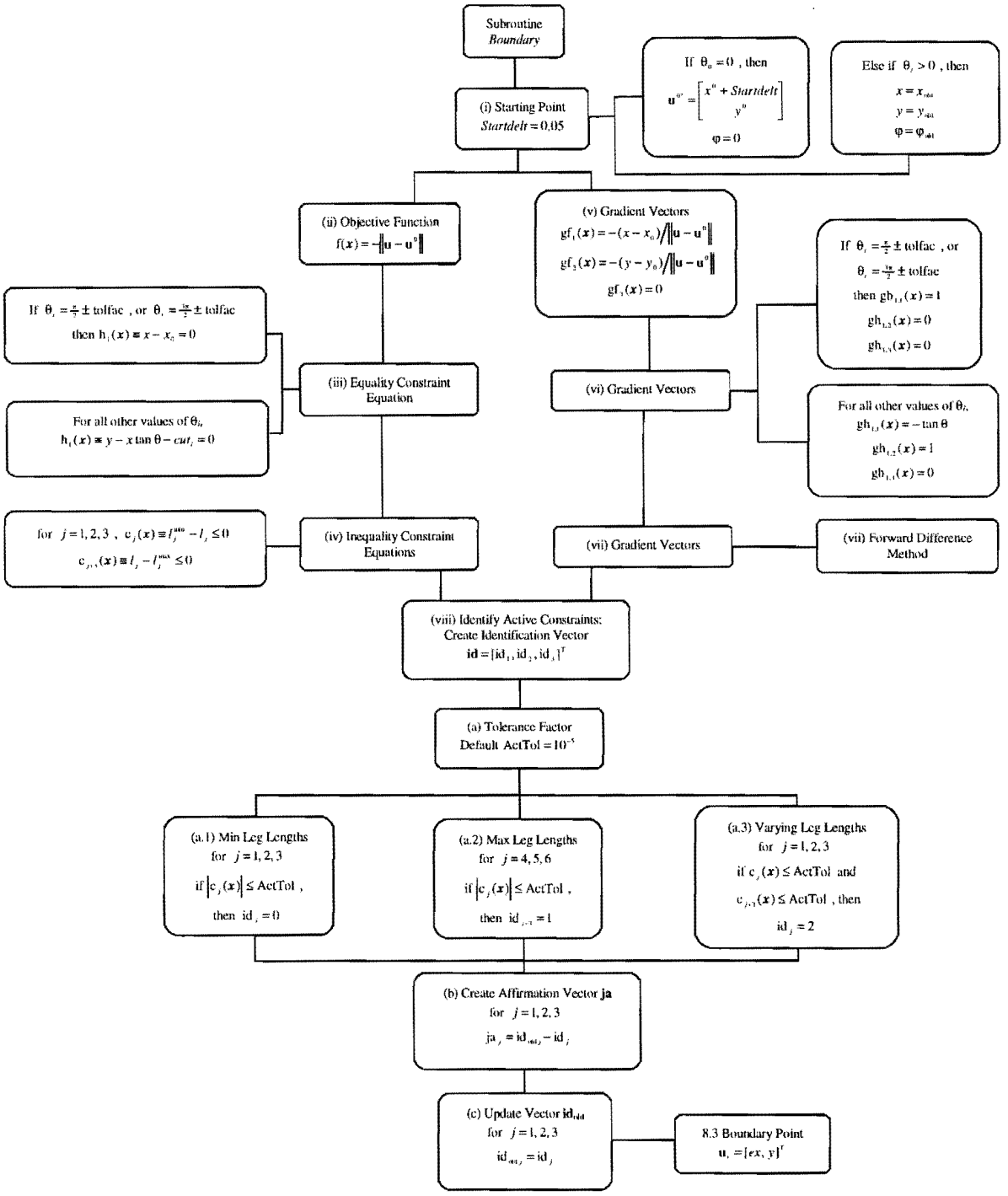

Figure A.3 Subroutine *Boundary*.

With $\theta_0 = 0$, an initial point on ∂A is sought, and optimization *problem (i)* as described in Section 2.3 is applicable. In order to make sure that the maximization is not done π out of phase, an offset (*Startdelt*) is added to the x-value of the radiating point $\mathbf{u}^0$. The actual radiating point used to find the initial point on ∂A with $\theta_0 = 0$, is $\mathbf{u}^{0*}$:

$$\mathbf{u}^{0*} = \begin{bmatrix} x^0 + Startdelt \\ y^0 \end{bmatrix} \tag{A.7}$$

With $\theta_i > 0$, the output and intermediate coordinates of the previous boundary point is used as an initial guess for the new boundary point sought.

Maximization *problem (i)* of Section 2.3 is converted to an equivalent minimization problem as follows:

$$\underset{\mathbf{u},\mathbf{w}}{minimize} \; -\left\|\mathbf{u}-\mathbf{u}^0\right\| \tag{A.8}$$

Analytical expressions for the gradient vectors of the objective function are used in subroutine *Boundary* as can be seen in box (vi) of Figure A.3.

The θ_i and Cut_i values determined in the main program, are used in subroutine *Boundary* to impose the single equality constraint. A separate equality constraint had to be defined to accommodate the asymptotic behavior of the tan-function:

$$\begin{aligned} &\text{If } \theta_i = \tfrac{\pi}{2} \pm TolFac \text{ or } \theta_i = \tfrac{3\pi}{2} \pm TolFac\text{, then: } \quad \mathrm{h}_1 \equiv x - x^0 = 0 \\ &\qquad\qquad \text{(default value } TolFac = 0.001\text{)} \\ &\text{for all other values of } \theta_i\text{: } \quad \mathrm{h}_1 \equiv y - x\tan\theta_i - Cut_i = 0 \end{aligned} \tag{A.9}$$

Analytical expressions for the gradient vectors of the equality constraint are used as can be seen in Figure A.3.

The minimum and maximum leg lengths that were entered in the main program, are used in subroutine *Boundary* for the six inequality constraint equations (see box (v) in Figure A.3).

$$\begin{aligned} \mathrm{c}_j &\equiv l_i^{\min} - l_i \le 0 \\ \mathrm{c}_{j+3} &\equiv l_i - l_i^{\max} \le 0 \end{aligned} \tag{A.10}$$

for $j = 1, 2, 3$

These inequalities impose correspond to expression (2.18) of Section 2.6.1. Once again, the forward difference method (A.3) is used to determine the gradient vectors of the inequality constraints.

An important aspect of subroutine *Boundary*, is to identify the active inequality constraints, as the mapping of the boundary is done. For each θ_i, $i = 0, 1, 2, \ldots, (\mathrm{Max}\,\theta\,\mathrm{Inc} - 1)$, the values of the

inequality constraints are monitored, and the entries of an identification vector, $\mathbf{id} = [\mathrm{id}_1, \mathrm{id}_2, \mathrm{id}_3]^T$, as well as the entries of an affirmation vector $\mathbf{ja} = [\mathrm{ja}_1, \mathrm{ja}_2, \mathrm{ja}_3]^T$ is determined. These vectors are used in the main program to identify the *bifurcation points*, as the workspace boundary is mapped.

A tolerance factor *ActTol* is associated with vector **id**, and its magnitude is specified by the user (default value $ActTol = 10^{-5}$). Each entry of the vector **id** can have one of three possible entries:

- if any of the actuator legs is at its minimum length, the corresponding entry in vector **id** will have the value *zero*.

$$\therefore \text{for } j = 1,2,3: \text{ if } |c_j| \le \text{ActTol}, \text{ then } \mathrm{id}_j = 0$$

- if any of the actuator legs is at its maximum length, the corresponding entry in vector **id** will have the value *one*.

$$\therefore \text{for } j = 4,5,6: \text{ if } |c_j| \le \text{ActTol}, \text{ then } \mathrm{id}_{j-3} = 1$$

- if any of the actuator legs is varying anywhere between its minimum and maximum length, the corresponding entry in vector **id** will have the value *two*.

$$\therefore \text{for } j = 1,2,3: \text{ if } c_j \le \text{ActTol } \textit{and } c_{j+3} \le \text{ActTol}, \text{ then } \mathrm{id}_j = 2$$

The entries of the affirmation vector (**ja**) is determined at each θ_i, $i = 0, 1, 2, \ldots, (\text{Max}\,\theta\,\text{Inc} - 1)$, by subtracting the current identification vector (**id** obtained for θ_i, $i = 1, 2, \ldots, \text{Max}\,\theta\,\text{Inc} - 1$) from the "old" identification vector ($\mathbf{id}_{\text{old}}$, which in actual fact is **id** obtained for θ_i, $i = 0, 1, 2, \ldots, \text{Max}\,\theta\,\text{Inc} - 2$).

$$\mathbf{ja} = \mathbf{id}_{\text{old}} - \mathbf{id} \tag{A.11}$$

With $\theta_0 = 0$, the initialized vector $\mathbf{id}_{\text{old}} = [0, 0, 0]^T$ is used to determine the affirmation vector (**ja**). After the affirmation vector has been determined, the vector $\mathbf{id}_{\text{old}}$ is updated, i.e. for $j = 1, 2, 3$, $\mathrm{id}_{\text{old}\,j} = \mathrm{id}_j$.

The coordinates of the "boundary point" ($\mathbf{u}^{bi} = [ex, y]^T$),the identification vector **id** as well as the affirmation vector **ja** are transferred back to the main program.

Each boundary bifurcation point is entered in a consecutive row *k* of the matrix **IDBound** using a similar notation to the one used for matrix **IDAll**. Once all the boundary bifurcation points are found, a comparison between the matrices **IDAll** and **IDBound** allows for the isolation of the bifurcation points

not situated on the accessible output set boundary. Counter k is initialized before the boundary mapping is started.

The main program uses the two vectors **id** and **ja** to identify the bifurcation points as the workspace boundary is mapped.

Clearly it is possible to intersect a bifurcation point with ray 0 where $\theta_0 = 0$, and provision is made to identify such a bifurcation point, if this happens.

with $\theta_0 = 0$, if: $\mathrm{id}_1 > 1$, *or* $\mathrm{id}_2 > 1$, *or* $\mathrm{id}_3 > 1$, then a bifurcation point *is not* intersected by ray 0.

with $\theta_0 = 0$, if: $\mathrm{id}_1 \leq 1$, *and* $\mathrm{id}_2 \leq 1$, *and* $\mathrm{id}_3 \leq 1$, then a bifurcation point *is* intersected by ray 0.

Increment counter $k = k + 1$

With $\theta_0 = 0$, the identification vector (**id**) shows whether each of the three actuator legs is at its maximum or minimum length.

With $\theta_i > 0$, it is the affirmation vector (**ja**) that indicates whether a bifurcation point is situated between two successively mapped boundary points:

with $\theta_i > 0$, if: $\mathrm{ja}_1 = 0$, *and* $\mathrm{ja}_2 = 0$, *and* $\mathrm{ja}_3 = 0$,

then a bifurcation point *is not* present between rays i and $i-1$.

with $\theta_i > 0$, if: $\mathrm{ja}_1 \neq 0$, *and* $\mathrm{ja}_2 \neq 0$, *and* $\mathrm{ja}_3 = 0$,

or $\mathrm{ja}_1 \neq 0$, *and* $\mathrm{ja}_2 = 0$, *and* $\mathrm{ja}_3 \neq 0$, ***or*** $\mathrm{ja}_1 = 0$, *and* $\mathrm{ja}_2 \neq 0$, *and* $\mathrm{ja}_3 \neq 0$,

then a bifurcation point *is* present between rays i and $i-1$.

Increment counter $k = k + 1$.

The detail of why any two entries of the affirmation vector **ja** has to be non-zero values to indicate bifurcation is evident from the discussion of the results in Section 2.6.3.1.

The main program creates a matrix called *Points* which has (MaxθInc) rows and four columns. The global x- and y-coordinates of mapped workspace boundary are respectively entered in columns 3 and 4 of matrix *Points* (see Box 8 in Figure 9). The global x- and y-coordinates of the mapped bifurcation points are respectively entered in columns 1 and 2 of matrix *Points*. This matrix is then used to create a drawing of the workspace (see Figure A.4).

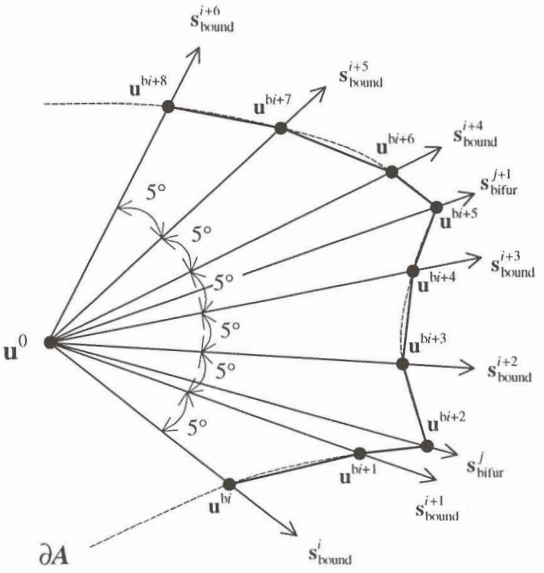

Figure A.4 Finding ∂A using *boundary* and *bifurcation* mappings.

If only the *boundary* mappings are used, the danger exists of inaccurately mapping the accessible output set as shown in Figure A.5.

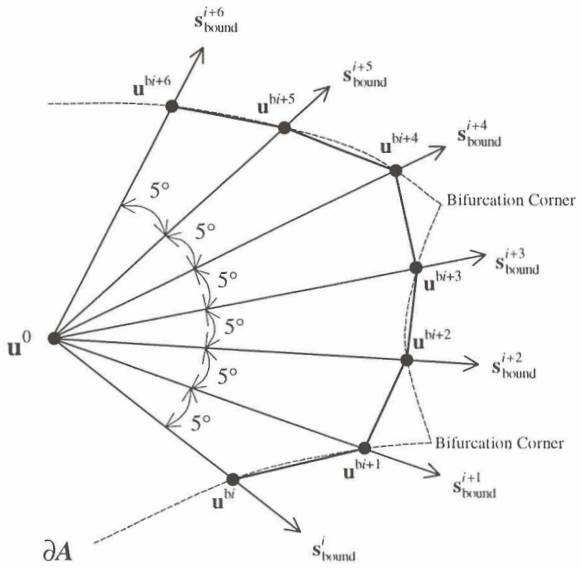

Figure A.5 Finding ∂A using only *boundary* mappings.

As soon as bifurcation point is identified between two boundary mappings, the specific coordinates of the bifurcation point is determined, and entered into columns 1 and 2 of matrix *Points*. The determination of the bifurcation point coordinates is done by interpretating the vector **id** (for $\theta_0 = 0$) *or*

the vector **ja** *and* the vector **id** (for $\theta_i > 0$) to determine the actual lengths of the actuator legs, and creating a vector containing these extreme actuator leg lengths $\boldsymbol{l}^{\text{ext}} = [l_1^{\text{ext}}, l_2^{\text{ext}}, l_3^{\text{ext}}]^{\text{T}}$. The vector $\boldsymbol{l}^{\text{ext}}$ is then used in subroutine *Bifurcation* to determine the coordinates of the bifurcation point.

If a bifurcation point is intersected by ray 0 where $\theta_0 = 0$, the vector $\boldsymbol{l}^{\text{ext}}$ is determined from the entries of the identification vector:

- extreme leg lengths corresponding to the minimum leg lengths:

$$\therefore \text{for } j = 1,2,3: \text{ if } \text{id}_j = 0, \text{ then } l_j^{\text{ext}} = l_j^{\min}$$

$$\text{IDBound}(k, j) = 0$$

- extreme leg lengths corresponding to the maximum leg lengths:

$$\therefore \text{for } j = 1,2,3: \text{ if } \text{id}_j = 1, \text{ then } l_j^{\text{ext}} = l_j^{\max}$$

$$\text{IDBound}(k, j) = 1$$

With $\theta_i > 0$, a bifurcation point is identified if any two entries of the affirmation matrix **ja** is a non-zero value. Since $\mathbf{ja} = \mathbf{idold} - \mathbf{id}$, two entries of the vector **id** change when a bifurcation point is present in the section of the boundary contained between the vectors **idold** and **id**. Mapping the *unchanged* leg lengths is done by examining the vector **id** as well as the vector **ja**.

- extreme leg lengths corresponding to the minimum leg lengths:

$$\therefore \text{for } j = 1,2,3: \text{ if } \text{id}_j = 0 \textit{ and } \text{ja}_j = 0, \text{ then } l_j^{\text{ext}} = l_j^{\min}$$

$$\text{IDBound}(k, j) = 0$$

- extreme leg lengths corresponding to the maximum leg lengths:

$$\therefore \text{for } j = 1,2,3: \text{ if } \text{id}_j = 1 \textit{ and } \text{ja}_j = 0, \text{ then } l_j^{\text{ext}} = l_j^{\max}$$

$$\text{IDBound}(k, j) = 1$$

Mapping the *changed* leg lengths is done by examining only the vector **ja**.

- maximum leg lengths *changing* to varying leg lengths:

$$\therefore \text{for } j = 1,2,3: \text{ if } \text{ja}_j = 1 - 2 = -1, \text{ then } l_j^{\text{ext}} = l_j^{\max}$$

$$\text{IDBound}(k, j) = 1$$

- minimum leg lengths *changing* to varying leg lengths:

$$\therefore \text{for } j = 1,2,3: \text{ if } \text{ja}_j = 0 - 2 = -2, \text{ then } l_j^{\text{ext}} = l_j^{\min}$$

$$\text{IDBound}(k, j) = 0$$

- varying leg lengths *changing* to maximum leg lengths:

$$\therefore \text{for } j = 1,2,3 : \text{if } \text{ja}_j = 2 - 1 = 1, \text{ then } l_j^{\text{ext}} = l_j^{\max}$$

$$\text{IDBound}(k, j) = 1$$

- varying leg lengths *changing* to minimum leg lengths:

$$\therefore \text{for } j = 1,2,3 : \text{if } \text{ja}_j = 2 - 0 = 2, \text{ then } l_j^{\text{ext}} = l_j^{\min}$$

$$\text{IDBound}(k, j) = 0$$

The vector $\boldsymbol{l}^{\text{ext}}$ is transferred by the main program to subroutine *Bifurcation* where the coordinates of the bifurcation points are determined (see Figure A.6).

A.2.3 Subroutine *Bifurcation*:

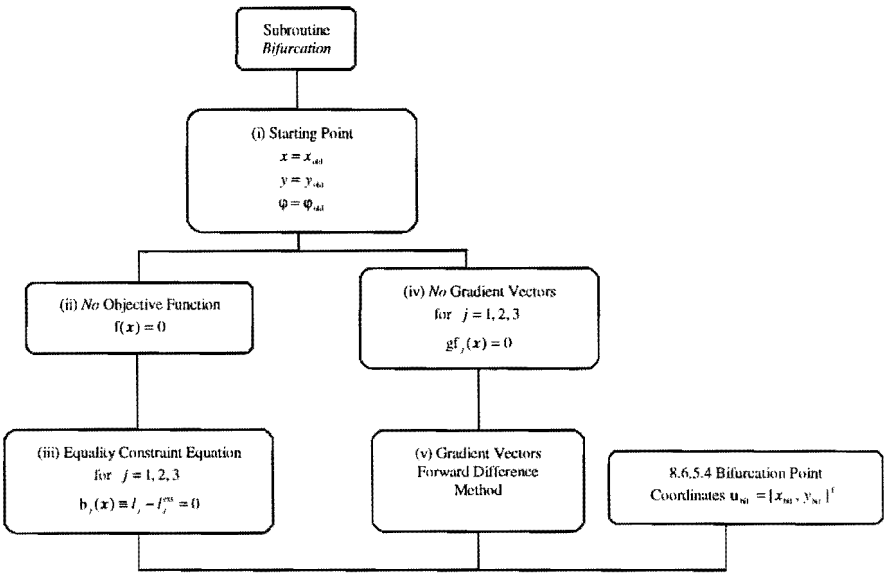

Figure A.6 Subroutine *Bifurcation*.

The starting point used in this subroutine is coordinates of the previous boundary point, and similar to subroutine *Start*, there is **no** explicit objective function, as well as **no** objective function gradient vectors for the code *LFOPCV3*.

The components of the vector $\boldsymbol{l}^{\text{ext}}$ are used in the three equality constraint equations shown in box (iv) of Figure A.6. *LFOPCV3* is once again used to solve three non-linear equations:

$$\begin{aligned} \text{v}_1(\mathbf{u}, \text{w}) - \text{v}_1^{\text{ext}} &= 0 \\ \text{v}_2(\mathbf{u}, \text{w}) - \text{v}_2^{\text{ext}} &= 0 \\ \text{v}_3(\mathbf{u}, \text{w}) - \text{v}_3^{\text{ext}} &= 0 \end{aligned} \qquad \text{(A.12)}$$

The gradient vectors of the equality constraints are determined using the forward difference method given by equation (A.3).

Once the coordinates of the bifurcation point ($\mathbf{u}^{bif} = [x^{bif}, y^{bif}]^T$) is determined, they are transferred to the main program, where they are respectively entered into columns 1 and 2 of the matrix *Points*.

Once the exterior boundaries are mapped, the bifurcation point connecting curves are traced using the matrices **IDAll** and **IDBound**. The first step in tracing the bifurcation point connecting curves is to identify the bifurcation points situated on the accessible output set boundary.

All row vectors of matrix **IDBound** is subsequently subtracted from each row vector in matrix **IDAll** to give the resultant vector **IDEA**:

$$\mathbf{IDEA} = \mathbf{IDAll} - \mathbf{IDBound} \tag{A.13}$$

If for any of the row vectors in matrix **IDBound** vector **IDEA** is a zero vector, the specific row vector in **IDAll** is labeled as it represents a bifurcation point situated on the boundary of the accessible output set. After the complete boundary is mapped, the unlabeled row vectors of **IDBound** is isolated and used to trace the bifurcation point connecting curves as described in Section 2.6.3.2 and set out in Figure A.2 and Figure A.7.

A.2.4 Subroutine Interior

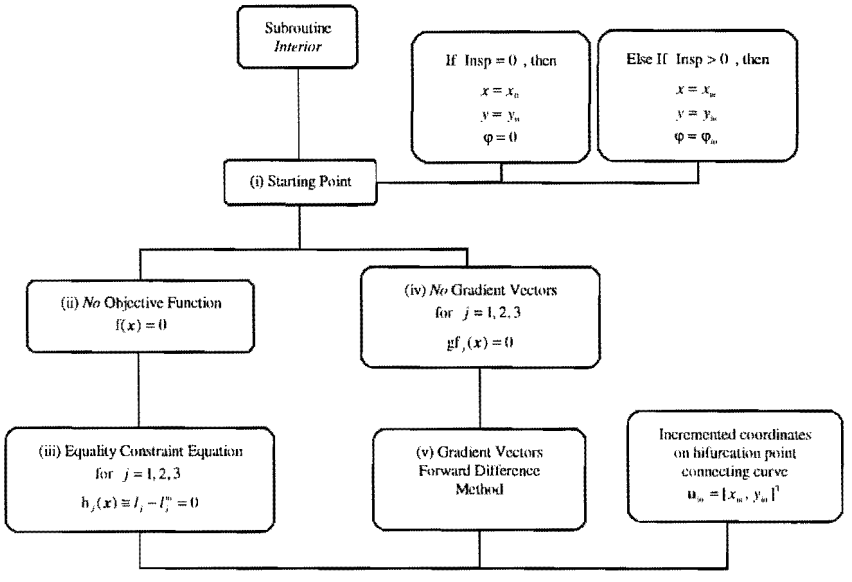

Figure A.7 Subroutine *Interior*

The starting point used in this subroutine is the radiating point $\mathbf{u}^0$ when the first point of a new bifurcation point connecting curve is to be traced. Once the first point on the curve is found, its coordinates are used as the starting point from where the next point on the curve is to be traced.

is coordinates of the previous boundary point, and similar to subroutine *Start* and *Bifurcation*, there is **no** explicit objective function, as well as **no** objective function gradient vectors for the code *LFOPCV3*.

The components of the vector $\boldsymbol{l}^{in}$ are used in the three equality constraint equations shown in box (iv) of Figure A.7. *LFOPCV3* is once again used to solve three non-linear equations:

$$\begin{aligned} v_1(\mathbf{u}, w) - v_1^{in} &= 0 \\ v_2(\mathbf{u}, w) - v_2^{in} &= 0 \\ v_3(\mathbf{u}, w) - v_3^{in} &= 0 \end{aligned} \tag{A.14}$$

The gradient vectors of the equality constraints are determined using the forward difference method given by equation (A.3).

Once the coordinates of the point on the interior curve ($\mathbf{u}^{in} = [x^{in}, y^{in}]^T$) is determined, they are transferred to the main program and entered into a *script* file from where the results are drawn.

This concludes the description of the computer code *PLANSTEW*.

Appendix B

B The Mapping of the Near Global Optimum Boundary Curves of the Reachable 6–3 Stewart Platform Workspace

The method for computing the accessible workspace for the 6–3 Stewart platform is explained further in this appendix, with the emphasis on the near global optimum boundary curves (*EF* $[11--11]$in Figure 3.3 and *DG* $[01--10]$ in Figure 3.4).

Following the "upward sweep" to map the reachable workspace as depicted in Figure 3.3, no problems occur as the near global optimum boundary curve *DE* $[-1111-]$ is mapped. Even for the first few rays mapping curve *EF* $[11--11]$, the near global maximum displacement from $\mathbf{u}^0$ is found time and again, and the first part of curve *EF* is easily determined as shown in Figure B.1. However, as curve *EF* is followed using the "upward sweep", the near global optimum is separated further and further from the global optimum situated along curve *FG* $[-1111-]$ (see Figure 3.2). As soon as the distance between the near global and global maximum displacements for two successive rays reaches a critical value, the optimizer *LFOPCV3* "jumps" to the global optimum for the latter ray. This explains the "jump" between the near global boundary curve *EF* $[11--11]$ and global boundary curve *FG* $[-1111-]$ as shown in Figure B.1.

Curve *EF* $[11--11]$ as presented in Figure 3.2 and Figure 3.3 is mapped with user interference. Because the first part of this curve is successfully determined, the optimization approach allows for the identification the actuator legs assuming extreme lengths as the working point follows curve *EF*. The label of curve *EF* $[11--11]$ stems from this identification procedure, and the label shows that actuator legs 1, 2, 5 and 6 remain fixed at their maximum lengths along curve *EF*.

Using a separate procedure, the complete near global optimum curve *EF* $[11--11]$ is mapped for N_R successive rays emanating in the range $121.8^\circ \leq \varphi_i \leq 180^\circ$ $(2.126 \leq \varphi_i \leq \pi)$, by minimizing the following error function using *LFOPCV3*.

$$e(\mathbf{u},\mathbf{w}) = \left(v_1(\mathbf{u},\mathbf{w}) - v_1^{\max}\right)^2 + \left(v_2(\mathbf{u},\mathbf{w}) - v_2^{\max}\right)^2 + \left(v_5(\mathbf{u},\mathbf{w}) - v_5^{\max}\right)^2 + \left(v_6(\mathbf{u},\mathbf{w}) - v_6^{\max}\right)^2 + (u_2 - u_1\tan 0)^2 + \left(\sqrt{u_{1j}^2 + u_{2j}^2} - (z^0 - u_{3j})\tan(\varphi_j)\right)^2 \quad \text{(B.1)}$$

The first four terms of (B.1) fixes the actuator legs at their extreme lengths while the fifth term fixes the vertical plane at $\theta_i = 0^\circ$. The last term of (B.1) corresponds to equation (3.15), and is incremented as curve EF $[11--11]$ is traced, i.e.

$$\varphi_j = 2.126 + \frac{j1.016}{N_R} \quad \text{(B.2)}$$

for $j = 0, 1, 2, \ldots, N_R$

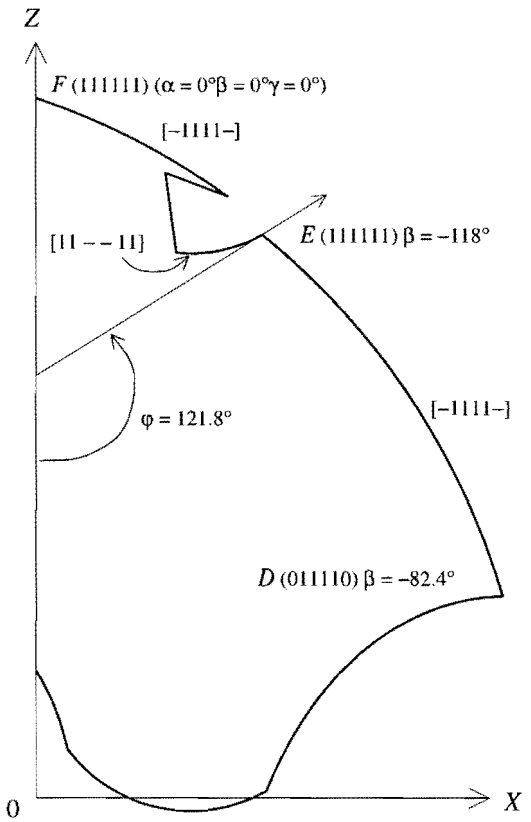

Figure B.1 The "jump" between the near global optimum and global optimum boundary curves.

Curve GD $[01--10]$ shown in Figure 3.4 is mapped in a similar manner.

Appendix C

C Procedure of Finding the Bifurcation Point Coordinates of the Fixed Orientation 6–3 Stewart Platform Workspace

As an extenuation of Section 4.3, an explanation follows on the details of how the bifurcation points on the boundary ($\partial A\ [0^\circ, 0^\circ, 0^\circ]$), of the fixed orientation workspace $A\ [0^\circ, 0^\circ, 0^\circ]$ of the 6–3 Stewart platform are mapped. Bifurcation point $B\ (0\ 1\ -\ -\ 1\ 0)\ [0^\circ, 0^\circ, 0^\circ]$ shown in Figure 4.1 and Figure 4.2(a) is used as a representative example.

Bifurcation point $B\ (0\ 1\ -\ -\ 1\ 0)\ [0^\circ, 0^\circ, 0^\circ]$ is found by minimizing an error function, corresponding to the planar case explained in Section 2.6.3.1, where an error function (2.24) was defined to find the point A' shown in Figure 2.10.

The error function used to find the coordinates of bifurcation point $B\ (0\ 1\ -\ -\ 1\ 0)\ [0^\circ, 0^\circ, 0^\circ]$, is again expressed in terms of the output coordinates (**u**) and intermediate coordinates (**w**).

Eight terms can be defined for the error function, one to "fix" the direction of the vertical plane (3.11), four terms to "fix" actuator legs 1, 2, 5 and 6 to their respective extreme lengths and a final three to "fix" the orientation of the top platform (4.4).

The error function of the planar Stewart platform is defined in terms of three coordinates (two output and one intermediate), and it consists of three terms. This means that for the spatial Stewart platform under consideration, the error function defined in terms of six variables should only have six terms.

Fortunately, not all eight of the possible terms are independent. The six terms used determine the orientation of the top platform (4.4) and any three of the four "active" actuator legs. Because the platform is symmetrical about the *X0Z* plane and all the leg length limits are the same, the fourth leg automatically assumes the correct extreme length at the correct orientation of the vertical plane.

Considering, for example, bifurcation point B $(0\ 1\ -\ -\ 1\ 0)$ $[0^\circ, 0^\circ, 0^\circ]$. It is expected that actuator legs 3 and 4 assume the same intermediate length when the other four legs assume the labeled extreme lengths. The error function to be minimized is:

$$e(\mathbf{u}, \mathbf{w}) = \left(v_1(\mathbf{u}, \mathbf{w}) - v_1^{\min}\right)^2 + \left(v_2(\mathbf{u}, \mathbf{w}) - v_2^{\max}\right)^2 + \left(v_5(\mathbf{u}, \mathbf{w}) - v_5^{\max}\right)^2 + (w_1 - 0)^2 + (w_2 - 0)^2 + (w_3 - 0)^2 \quad \text{(C.1)}$$

The global coordinates that were obtained for bifurcation point B $(0\ 1\ -\ -\ 1\ 0)$ $[0^\circ, 0^\circ, 0^\circ]$, by minimizing the error function (B.1) using *LFOPCV3*, are $(6.197,\ 0.0,\ 6.613)$.

These coordinates and the fixed orientation of the top platform ($\alpha = 0^\circ, \beta = 0^\circ$ and $\gamma = 0^\circ$), are substituted into expressions (3.3) and (3.4) to determine the actuator leg lengths:

$$l_1 = l_6 = 8.0 = l_{\min}$$
$$l_2 = l_5 = 15.0 = l_{\max}$$
$$l_3 = l_4 = 11.331$$

This validates the definition of error function (C.1).

Appendix D

D Determination of a Non-Vertical Bifurcation Curve of the Fixed Orientation 6–3 Stewart Platform Workspace

Bifurcation line $A'B'C'$ $[0^\circ, 0^\circ, -30^\circ]$ presented as part of the fixed orientation accessible workspace boundary ∂A $[0^\circ, 0^\circ, -30^\circ]$ shown in Figure 4.3 is analyzed in more detail looking at the three vertical planes of the fixed orientation accessible workspace A $[0^\circ, 0^\circ, -30^\circ]$, respectively isolated at $\theta = 15^\circ$, $\theta = 30^\circ$ and $\theta = 45^\circ$.

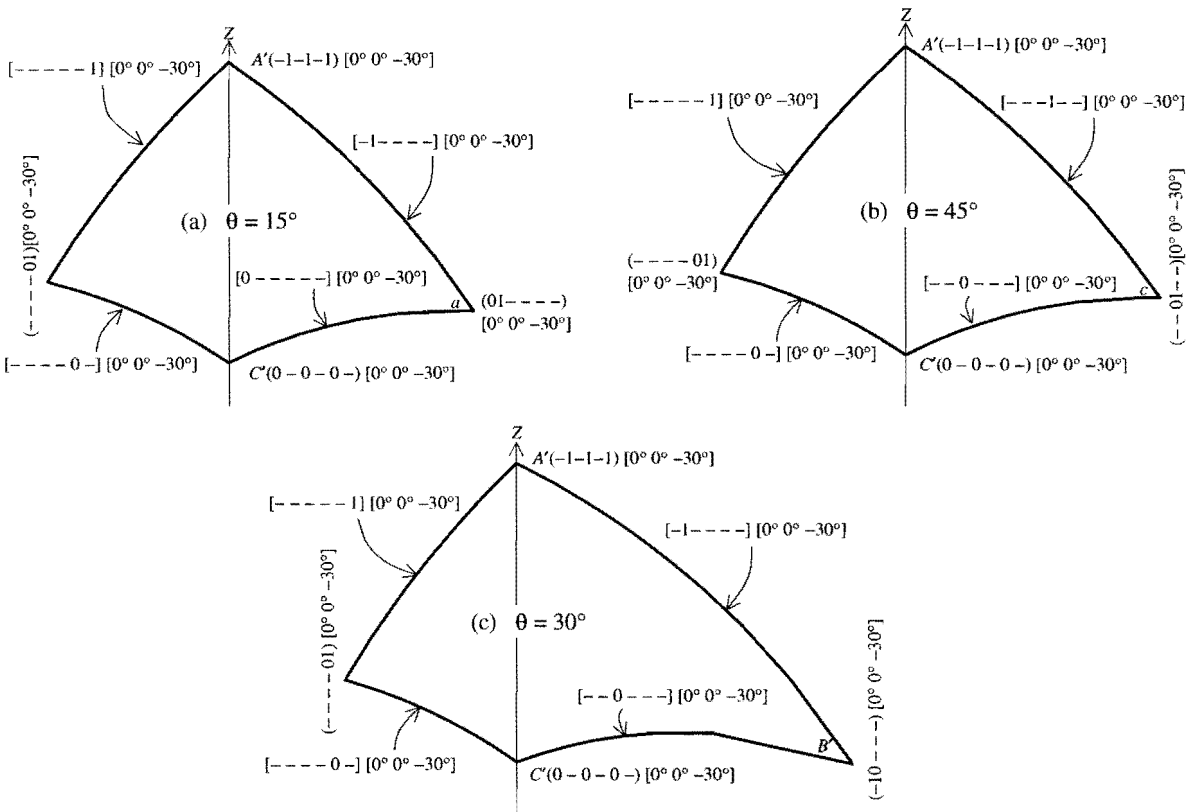

Figure D.1 Sections of ∂A $[0^\circ, 0^\circ, -30^\circ]$ at (a) $\theta_i = 15^\circ$, (b) $\theta_i = 45^\circ$, and (c) $\theta_i = 30^\circ$.

Figure D.1 (a) shows the section of the fixed orientation accessible boundary ∂A $[0^\circ, 0^\circ, -30^\circ]$ in the vertical plane at $\theta_i = 15^\circ$. Along curve $A'a$ $[-1 - - - -]$ $[0^\circ, 0^\circ, -30^\circ]$ $[\theta_i = 15^\circ]$, actuator leg 2 remains at its maximum length as the manipulator working point follows the curve. Curve $A'a$ forms

part of the convex boundary surface $A'B'D'$ labeled in Section 4.3 as $A'B'D'$ $[-\ 1\ -\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$. Similarly curve aC' $[0\ -\ -\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$ $[\theta_i = 15^\circ]$ is characterized by actuator leg 1 remaining fixed at its minimum length, and curve aC' forms part of the concave boundary surface $C'B'D'$ $[0\ -\ -\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$.

Next consider Figure D.1 (b) which shows the section of the fixed orientation accessible boundary $\partial A\,[0^\circ,\ 0^\circ,\ -30^\circ]$ in the vertical plane at $\theta_i = 45^\circ$. The labels of convex curve $A'b$ $[-\ -\ -\ 1\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$ $[\theta_i = 45^\circ]$ and concave curve bC' $[-\ -\ 0\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$ $[\theta_i = 45^\circ]$ respectively correspond to the labels of convex boundary surface $A'E'B'$ and concave boundary surface $C'E'B'$ labeled in Section 4.3.

The *actual* bifurcation curve $A'B''C'$ coinciding with the intersection of the four boundary surfaces $A'B'D'$ and $A'E'B'$, as well as $C'B'D'$ and $C'E'B'$ (as labeled in Section 4.3), consists out of two bifurcation lines. The upper bifurcation line is a convex, and is labeled $A'B''$ $[-\ 1\ -\ 1\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$, while the bottom bifurcation line is and a concave with label $B''C'$ $[0\ -\ 0\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$. These two bifurcation lines intersect at bifurcation point B'' $(0\ 1\ 0\ 1\ -\ -)$ $[0^\circ,\ 0^\circ,\ -30^\circ]$.

The curve $A'B'C'$ obtained from the vertical plane at $\theta_i = 30^\circ$ is labeled and shown in Figure D.1 (c). Convex "bifurcation line" $A'B'$ $[-\ 1\ -\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$ $[\theta_i = 30^\circ]$ does in actual fact not coincide with the intersection of boundary surfaces $A'B'D'$ and $A'E'B'$, as it is part of boundary surface $A'B'D'$ $[-\ 1\ -\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$. Similarly, concave "bifurcation line" $B'C'$ $[-\ -\ 0\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$ $[\theta_i = 30^\circ]$ forms part of boundary surface $C'E'B'$ $[-\ -\ 0\ -\ -\ -]$ $[0^\circ,\ 0^\circ,\ -30^\circ]$, and does not coincide with the intersection of the two boundary surfaces $C'E'B'$ and $C'B'D'$.

The actual bifurcation curve $A'B''C'$ is shown in Figure D.2 together with the near bifurcation curve $A'B'C'$ and they almost coincide.

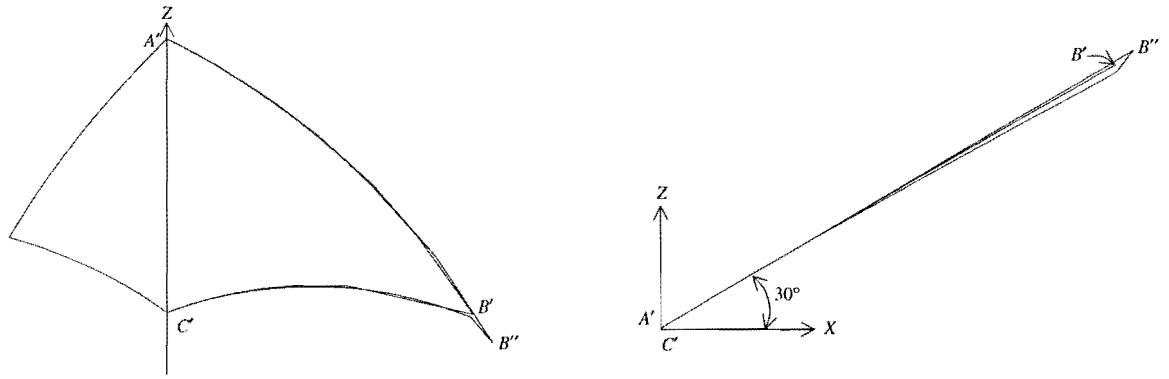

Figure D.2 The near ($A'B'C'$) and actual ($A'B''C'$) bifurcation curves.

The computed "bifurcation" curves $A'B'C'\left[0^\circ, 0^\circ, -30^\circ\right]$, $A'D'C'\left[0^\circ, 0^\circ, -30^\circ\right]$, $A'E'C'$ $\left[0^\circ, 0^\circ, -30^\circ\right]$, $A'F'C'$ $\left[0^\circ, 0^\circ, 30^\circ\right]$, $A'G'C'$ $\left[0^\circ, 0^\circ, 30^\circ\right]$ and $A'H'C'$ $\left[0^\circ, 0^\circ, 30^\circ\right]$ as presented in Figure 4.4 are also not the exact bifurcation curves. The deviations are however sufficiently small so as to be considered negligible from a practical point of view.

In addition the bifurcation lines all lie outside the dextrous workspace $\boldsymbol{A}\left[0^\circ, 0^\circ, (-30^\circ)-(30^\circ)\right]$ (Volume $A'C'I'J'K'L'M'N'$) as shown in Figure 4.5 and may therefore be ignored in the further analysis of the dextrous workspace.