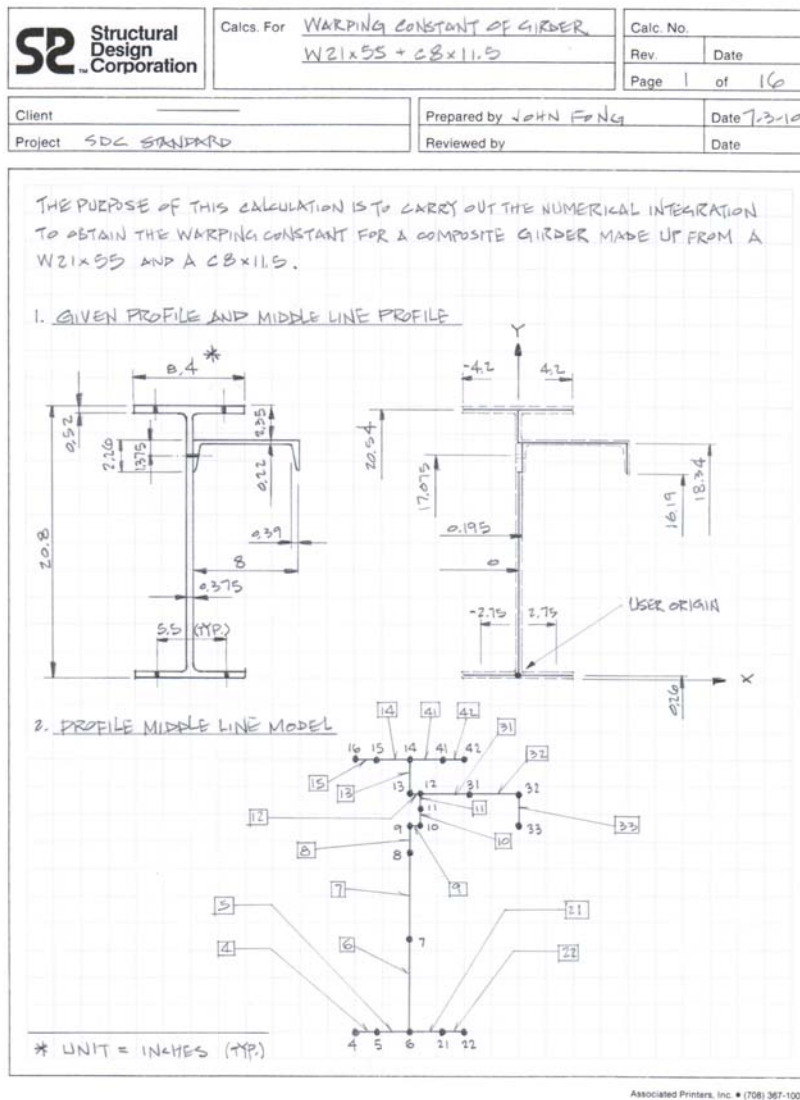


# Warping Constant of Open Sections with Arbitrary Profile Geometry

John Fong (馮永康) . Bill Vanni  
Structural Design Corporation  
1133 Claridge Dr., Libertyville, IL 60048  
©July-August 2010, Rev. 1 ©November 2011


As it appears in several editions of AISC “Steel Construction Manual”, a close encounter with warping constant ( $C_w$ ) is unavoidable when dealing with buckling and/or torsion behavior of structural members. Formulas for calculating warping constant for members with standard profile geometry are readily available from many references. There should be no problem for most structural members with mono- or doubly- symmetric sections in general. For members with a cap channel over an I-beam (and collectively many other standard sections), a good summary of information for torsional properties can be found in “Torsional Section Properties for Steel Shapes”, Canadian Institute of Steel Construction. The problem is how to calculate  $C_w$  for sections with non-standard (or arbitrary) profile geometry.



Albeit there is no shortage of advisory materials on torsion theory but they were often accompanied by many not-so-user-friendly formulas found in many textbooks.

This article serves only one purpose. It is not about understanding the theory behind warping constant nor is it about deriving complicated equations for torsion or offering their closed-form solutions. By being called a “**practical** reference” here, we mean a **comprehensive numerical example** that would:

- explicitly dissect all the numerical integration in steps and carry out the necessary calculations leading to  $C_w$  and
- in the process, establish a standard solution procedure and methodology applicable to any thin-walled open section.

	Calcs. For <u>WARPING CONSTANT OF GIRDER</u>		Calc. No.	
	<u>W21X55 + C8X11.5</u>		Rev.	Date
			Page <u>2</u> of <u>10</u>	
Client _____		Prepared by <u>JOHN FONG</u>		Date <u>7-3-10</u>
Project <u>SDC STANDARD</u>		Reviewed by _____		Date _____

3. NODE AND SEGMENT GEOMETRY

N O D E			S E G M E N T							
ID	USER COORDINATES		ID	NODE		THICKNESS [t]	FROM		TO	
	X	Y		FROM <sub>i</sub>	TO <sub>j</sub>		X <sub>i</sub>	Y <sub>i</sub>	X <sub>j</sub>	Y <sub>j</sub>
4	-4.2	0.26	10	10	11	0.765	0.195	16.19	0.195	17.075
5	-2.75	0.26	11	11	12	0.765	0.195	17.075	0.195	18.34
6	0	0.26	12	12	13	0.001*	0.195	18.34	0	18.34
7	0	5.2	13	13	14	0.375	0	18.34	0	20.54
8	0	12.8	14	14	15	0.52	0	20.54	-2.75	20.54
9	0	16.19	15	15	16	0.52	-2.75	20.54	-4.2	20.54
10	0.195	16.19	21	6	21	0.52	0	0.26	2.75	0.26
11	0.195	17.075	22	21	22	0.52	2.75	0.26	4.2	0.26
12	0.195	18.34	31	12	31	0.22	0.195	18.34	4	18.34
13	0	18.34	32	31	32	0.22	4	18.34	7.9925	18.34
14	0	20.54	33	32	33	0.39	7.9925	18.34	7.9925	16.19
15	-2.75	20.54	4	4	5	0.52	-4.2	0.26	-2.75	0.26
16	-4.2	20.54	41	14	41	0.52	0	20.54	2.75	20.54
21	2.75	0.26	42	41	42	0.52	2.75	20.54	4.2	20.54
22	-4.2	0.26	5	5	6	0.52	-2.75	0.26	0	0.26
31	4	18.34	6	6	7	0.375	0	0.26	0	5.2
32	7.9925	18.34	7	7	8	0.375	0	5.2	0	12.8
33	7.9925	16.19	8	8	9	0.375	0	12.8	0	16.19
41	2.75	20.54	9	9	10	0.001*	0	16.19	0.195	16.19
42	4.2	20.54								

\*FITTING THICKNESS

Our goal here is to give ourselves a “Homework Problem” that calculates the  $C_w$  value for a “composite profile”. The profile is made from a W21X55 shape with a C8X11.5 channel, which is attached off the web of W21 at 3.5 inches below the W21 top flange extreme fiber.

For easy reference, all the calculation (Calc) pages prepared manually for this homework problem are scanned in as part of this article. It takes seventeen (17) steps to complete the task, hopefully without hiding much of the technical fine points. “Seventeen” is not a sacred number. This task can take a greater number of or fewer steps to suit our needs. It all depends on (a)

how familiar we are with the subject, (b) how we organize the presentation, and (c) how much detail we want to expose, add, hide or omit.

**Let us get started now:** Our example profile is taken after a real-life girder from a mill building. It looks innocent and does not appear much different from some of the classic profiles that we are all familiar with: “a cap channel over an I-beam”, which exists in so many industrial facilities except that our channel is a few inches off from both (the major and minor) axes.

We recommend that you now **take a break** from reading any further but skim through these Calc pages briefly to get some ideas, especially if you are just **curious** about why it takes so many steps to figure out a “**simple**”  $C_w$  constant for members with such “**simple**” profile geometry.

As we go through with steps (and pages) of the Calc in the ensuing paragraphs, some subject(s) that appear trivial or self-explanatory will be bypassed for obvious reasons. Meanwhile we will highlight a few tips or traps here and there wherever appropriate. Anyhow, if you are new to this **attempt**, please pay closer attention to the “**Process**” in problem solving techniques rather than in numerical details.

### Middle Line Profile $C_w$ Model

Our “ $C_w$  model” found on Calc page 1 resembles a typical plane frame model employed in regular structural analysis. The so-called “**middle line**” suggests that all the framework linkages in the model would always traverse along the **line of mid-thickness** of each and every profile component regardless to how thick each may be. Middle line model is an **idealized** skeleton profile. It is nothing but a graphical description of the general element “geometry” and the element “connectivity”.

For profiles with **plain** elements, their design is mostly controlled by “strength requirement”. Many rolled shapes having simple flanges and/or webs fit in this category. Constructing a middle line model for these profiles is straightforward as if drawing elements by connecting the dots. Take a plain I-beam for instance, we would “idealize” it with nominal six (6) nodes to depict (or **digitize**) all the participating components, which include two (2) flange elements and one (1) web element.


For profiles with much more **complex** geometry, their configuration can be dictated by “strength requirement” and/or “functionality”. But on occasions “**detailing constraint**” may play an important role as well. Complex profile geometries are found regularly in cold-formed or built-up sections. To facilitate fabrication, some of the elements are connected to one or more “adjacent element(s)” by welding or bolting. Here is how detailing constraint comes into play: the connections are made at strategic points usually at or near element’s terminal ends or any interior point to suit the bolt gage dimensions, etc. Therefore when identifying a “profile element”, it no longer applies to plain flanges and webs only. In a broader sense it should also include composite web(s), attachment(s) connected (by welds or bolts) to other flange(s) and/or web(s).

For any given profile, the value of its  $C_w$  **remains the same** whether we derived that value from complex or simple modeling. If our goal is **limited** to only obtaining warping constant and nothing else, we should go for the simplest “model construct and there is no need to worry about “detailing constraints”.

If our interested is not only the warping constant but also the state of stress of the profile then we must **consider** taking in pertinent “details” as part of our modeling strategy. To authenticate a profile middle line model, it would be more **persuasive** if we had included additional nodes at strategic points, where detailing constraints play a part. The implication of “details” is that we have to subdivide each affected elements into as many “sub-elements or segments” as required to cover all bases. Don’t be too skimpy over this because adding a few extra nodes “**before**” is always much better than “**after**”.

The “decision” of using extra/fewer nodes beyond “detailing constrains” can be influenced by other factors as well. Let’s look at a “**thrust-plated crane girder**” (thinking beyond  $C_w$ ) in that the top flange acts as a “flange” in resisting vertical loads. The same flange can also “act as web” when subjected to lateral loads. This analogy applies to other elements in similar or in reverse logic. Due to the double roles each element may play under different load natures, relying on results from using simpler model to justify the adequacy of the girder can be too slack for a few reasons:

- (a) the distribution pattern of (flexural  $\pm$  torsional) **shear** stress and (flexural bending  $\pm$  warping) **fiber** stress throughout the profile may be more complex than normally anticipated especially when dealing with many load combinations,
- (b) the allowable bending stress and allowable shear stress may vary throughout the profile especially when fatigue is one of the design requirements.



Calcs. For WARPING CONSTANT OF GIRDER

W21X55 + C8X11.5

Calc. No. \_\_\_\_\_

Rev. \_\_\_\_\_ Date \_\_\_\_\_

Page 3 of 16

Client \_\_\_\_\_

Project SDC STANDARD

Prepared by JOHN FONG

Reviewed by \_\_\_\_\_

Date 1-3-0

Date \_\_\_\_\_

**4. LOCATION OF CENTROID (C.G.)**

$$\begin{cases} L_x = x\text{-LENGTH PROJECTION} = [x_2] - [x_0] \\ L_y = y\text{-LENGTH PROJECTION} = [y_2] - [y_0] \\ L = \text{SEGMENT LENGTH} = ([L_x]^2 + [L_y]^2)^{1/2} \\ A = \text{SEGMENT AREA} = [L][t] \end{cases}$$

$$\begin{cases} X_c = \text{SEGMENT X-CENTROID} = ([x_1] + [x_2])/2 \\ Y_c = \text{SEGMENT Y-CENTROID} = ([y_1] + [y_2])/2 \\ A_x = \text{FIRST MOMENT X} = [A][X_c] \\ A_y = \text{FIRST MOMENT Y} = [A][Y_c] \end{cases}$$

SEGMENT ID	S E G M E N T   P R O P E R T I E S							
	$L_x$	$L_y$	$L$	$A$	$X_c$	$Y_c$	$A_x$	$A_y$
10	0	0.885	0.885	0.677	0.195	16.6325	0.132	11.2606
11	0	1.265	1.265	0.9677	0.195	17.7075	0.1887	17.136
12	-0.195	0	0.195	0.0002	0.0975	18.34	0	0.0036
13	0	2.2	2.2	0.825	0	19.44	0	16.038
14	-2.75	0	2.75	1.43	-1.375	20.54	-1.9662	29.3722
15	-1.45	0	1.45	0.754	-3.475	20.54	-2.6201	15.4872
21	2.75	0	2.75	1.43	1.375	20.54	1.9662	0.3718
22	1.45	0	1.45	0.754	3.475	20.54	2.6201	0.196
31	3.805	0	3.805	0.8371	2.0975	18.34	1.7558	15.3524
32	3.9925	0	3.9925	0.8783	5.9963	18.34	5.2668	16.1089
33	0	-2.15	2.15	0.8385	7.9925	17.265	6.7017	14.4767
4	1.45	0	1.45	0.754	-3.475	0.26	-2.6201	0.196
41	2.75	0	2.75	1.43	1.375	20.54	1.9662	29.3722
42	1.45	0	1.45	0.754	3.475	20.54	2.6201	15.4872
5	2.75	0	2.75	1.43	-1.375	0.26	-1.9662	0.3718
6	0	4.94	4.94	1.8525	0	2.73	0	5.0573
7	0	7.6	7.6	2.85	0	9	0	25.65
8	0	3.39	3.39	1.2713	0	14.495	0	18.4208
9	0.195	0	0.195	0.0002	0.0975	16.19	0	0.0032

In predicting the unpredictable (through **experience** or gut feeling), it is better to add a few extra nodes for girders with thrust plate. Better than otherwise these nodes should be located at strategic points that you “believe” would be helpful in evaluating the “state of stress” of your girders.

Just be aware that adding extra nodes to the model does not make much difference if you calculate  $C_w$  through automation. But it does become much more taxing if  $C_w$  is obtained by manual means.

As a result of “thinking beyond  $C_w$ ”, instead of using the bare minimum of 12 nodes, we make up our “middle

line profile model” with 20 “nodes” and 19 “node linkages” in total. Here we added:

- nodes 15 and 41 in the top flange to account for rail clip connection,
- node 11 right at the bolt gage location where it joins the web of W21 to one of the flanges of C8,
- nodes 5 and 21 in the bottom flange for seat bolt connection plus
- a few extra interior nodes.

As for naming the “linkages” between “nodes” in this article, we will purposely call them “**segments**” instead of “**elements**” considering the fact that one element may contain one or more subsets joined from a single or a multitude of segments.



## User Coordinate Axes System

This is the x-y Cartesian system chosen by the user for the convenience in “describing” profile geometry. Calling it the “user axes system” is to distinguish it from the “principal axes system” that we would establish shortly. The “orientation” and the “origin” of user axes system can be chosen arbitrarily. The liberty in choices does not apply to principal axes system for it is unique to a given profile.

	Calcs. For <u>WARPING CONSTANT OF GIRDER</u> <u>W21x55 x CB x 11.5</u>		Calc. No. _____ Rev. _____ Date _____ Page <u>4</u> of <u>16</u>	
	Client _____ Project <u>SDC STANDARD</u>		Prepared by <u>JOHN FONG</u> Date <u>7-3-10</u> Reviewed by _____ Date _____	
	<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SA = SUM OF AREA = <math>\sum [A] = 19.7338</math></p> <p>SAX = SUM OF <math>[Ax] = \sum [Ax] = 14.0451</math></p> <p>SAY = SUM OF <math>[Ay] = \sum [Ay] = 130.3679</math></p> </div> <div style="width: 45%;"> <p><math>Cx = x \text{ CENTROID} = [SAX]/[SA] = 0.7117</math></p> <p><math>Cy = y \text{ CENTROID} = [SAY]/[SA] = 11.6737</math></p> </div> </div> <p><u>5. GENERALIZED DOUBLE SECTORIAL AREA OF A SEGMENT</u></p> <p>FOR A SEGMENT CONNECTING NODE <math>i</math> AND NODE <math>j</math>, THE STRAIGHT LINE EQUATION IN THE x-y PLANE TAKES THE FORM:</p> $Ax + By = C \quad \text{WHERE} \quad \begin{cases} A = y_j - y_i \\ B = x_i - x_j \\ C = x_i y_j - x_j y_i \end{cases}$ <p><math>P</math> = NORMAL (PERPENDICULAR) DISTANCE FROM NODE <math>k</math> TO LINE <math>ij</math></p> $= \pm \frac{Ax_k + By_k - C}{(A^2 + B^2)^{1/2}}$ <p>WHERE THE <math>\pm</math> SIGN CORRESPONDS TO VECTORIAL NORMAL USING COUNTERCLOCKWISE RIGHT-HAND RULE FROM <math>i</math> TO <math>j</math> ABOUT NODE <math>k</math> POINTING DOWNWARD.</p> <p><math>A_0</math> = SECTORIAL AREA OF SEGMENT <math>ij</math> W.R.T. <math>k</math> <span style="float: right;"><math>= PL/2</math> <math>= Pds/2</math></span></p> <p><math>W</math> = DOUBLE SECTORIAL AREA W.R.T. <math>k</math></p> $= 2A_0 = Pds$ $= PL$ $= \frac{Ax_k + By_k - C}{(A^2 + B^2)^{1/2}} \cdot L \quad \text{WHERE } L = (A^2 + B^2)^{1/2}$ $= \frac{(y_j - y_i)x_k - (x_i - x_j)y_k + x_i y_j - x_j y_i}{(A^2 + B^2)^{1/2}} \cdot L$ $= \frac{-L y_j x_k + L x_j y_k + x_i y_j - x_j y_i}{(A^2 + B^2)^{1/2}} \cdot L$ $= \frac{-(L y_j x_k - L x_j y_k - x_i y_j + x_j y_i)}{(A^2 + B^2)^{1/2}} \quad \leftarrow \text{NODE } k \text{ MAY BE CENTROID OR SHEAR CENTER}$			

Associated Printers, Inc. • (708) 367-1009

Before locating the “origin” of the user system, let us define the “y-axis” to be collinear with the centerline of the web of W21. As for the “x-axis”, we prefer to have it lined up with the bottom extreme fiber of the bottom flange. You can verify on Calc page 1 the location of the “user system origin” based on this scheme. You may have it any other ways per your convenience but here are some of the advantages (or disadvantage) of our preference:

- (a) the y-coordinate for all the nodes would always be positive that we should appreciate later for not having to mess with so many numbers with negative sign many times over;
- (b) it would facilitate the re-assignment of nodal y-coordinates due to any revision in segment attributes especially changing in bottom flange thickness.

## Data Input

Notice that it requires seven (7) discrete entities of (input) information to describe the profile geometry. These input information (data) whether symbolic (text), logical, or numerical are **normalized** into two **core** tables:

- (a) **Node** table with three (3) fields for: node identification (ID), nodal x- and y-coordinates,

- (b) **Segment** table with four (4) fields for: Segment ID, “Incidence From” node ID, “Incidence To” node ID, and the material thickness.

In this application: all the IDs are of **symbolic** type while nodal coordinates and segment thickness are of **numerical type**. You may need to add extra “fields” in each table or create a few more tables to store your transition data as required.

Now let us review both Calc page 1 and page 2 and take notice of a few facts on our “Data Input”:

- (a) from a “**database design**” point of view, “**Node**” table takes higher precedence over “**Segment**” table. In other words, “Node” table (both the data structure design and data content input) should be established first.
- (b) the **numbering** system for both “Node” and “Segment” IDs is entirely **arbitrary**. We have choices of using “numerals” or “text characters”. Professionally we should use “text characters” for purpose of “identification” (not for “calculation”).
- (c) as for the “segment termination incidences”, there is no set pattern on which end should be the “From” node or which should be the “To” node. One of the rules we must adhere to is that any (input) node ID used for specifying segment connectivity purpose should already be defined in the “Node table” (thereby maintaining the **Data Referential Integrity**);
- (d) typical segment should always follow a rule to have uniform thickness between its “From” node and “To” node. In dealing with change in thickness within a segment, we must add extra node(s) in between making all segments comply with the thickness rule.
- (e) material thickness for segments joined by weld or bolt should be **additive** from all the participating segments. Using the “resulting thickness sum” is to simulate that these segments are “**fused**” into one idealized (**combined**) segment.

For instance: web thickness of W21 = 0.375 and flange thickness of C8 = 0.39. For segments 10 and 11 being joined by bolt at node 11, we would use combined thickness = 0.765 replicating that the W21 web and C8 flange would react in unison under loads;


- (f) for any “fused segment” **transitioning** contiguously to the immediate adjacent “clean-and-free-from-joining element”, the thickness should be considered “**insignificant**”. Such “segment” does not really exist. Its presence is only for model “linkage purpose” therefore we may assign a value that is relatively trivial comparing to the thickness of all other “**significant**” segments in the profile. As in our example for segments 9 and 12: we would assign a fictitious thickness = 0.001.

## Data Presentation

The “usefulness” of calculation is measured primarily by the “**numerical accuracy**”. Numerical mistakes found in engineering calculation are very much avoidable but inevitable because not every Preparer is perfect. This is why some calculations must be reviewed for quality assurance purpose. The bottom line is to catch the mistake and fix it in time. The amount of time it takes for a Reviewer to “review” depends somewhat on how **appealing** the information was arranged. If the “presentation” were clear to the Preparer, then it would take much less effort for the Reviewer to “identify” and “rectify” the mistake(s) should there be any.

Engineering calculation is a presentation of processes. Each process follows a preset routine. When we design a beam, the process may involve writing down dimensions, setting up applied loads, calculating internal forces and moments, applied stresses and allowable stresses, etc. Each beam is treated as a separate entity. The documentation for designing a beam follows a top-down line-by-line style. In calculating warping constant, the process also follows a top-down style but with a twist. It involves many cycles of marching **forward** for new data then referencing **backward** for existing data much as in a zigzag. In addition, to carry out each step the entire data set gets

involved. In other words, if we execute nodal (segment) data, information related all 20 nodes (19 segments) would get the same **mass-production** treatment.

	Calcs. For	WARPING CONSTANT OF GIRDER W21x55 + C3x11.5	Calc. No.	
			Rev.	Date
			Page	5 of 16
Client			Prepared by	John Folks
Project	SDC STANDARD		Reviewed by	
			Date	7-3-10

6. SEGMENT MOMENT OF INERTIA AND DOUBLE SECTIONAL AREA W.R.T. CENTROID

$$I_{xx} = [A] [(y_i - C_y)^2 + (y_i - C_y)(y_j - C_y) + (y_j - C_y)^2] / 3$$

$$I_{yy} = [A] [(x_i - C_x)^2 + (x_i - C_x)(x_j - C_x) + (x_j - C_x)^2] / 3$$

$$I_{xy} = [A] \{ [(x_i - C_x)(y_i - C_y) + (x_j - C_x)(y_j - C_y)] / 3 + [(x_i - C_x)(y_j - C_y) + (x_j - C_x)(y_i - C_y)] / 6 \}$$

SEGMENT ID	SEGMENT MOMENT OF INERTIA			NORMAL DISTANCE TO CENTROID = f	L	p ds
	I <sub>xx</sub>	I <sub>yy</sub>	I <sub>xy</sub>			
4	98.2263	13.3487	36.0308	11.4137	1.45	16.5499
5	186.2913	7.128	34.0588	11.4137	2.75	31.3878
6	151.95	0.9384	11.7921	-0.7117	4.94	-3.5159
7	34.0925	1.4437	6.4235	-0.7117	7.6	-5.4091
8	11.3359	0.644	-2.5526	-0.7117	3.39	-2.4128
9	0.004	0.0001	-0.0005	-4.5163	0.195	-0.8807
10	16.6917	0.1808	-1.7348	-0.5167	0.885	-0.4573
11	35.3602	0.2584	-3.0172	-0.5167	1.265	-0.6537
12	0.0087	0.0001	-0.0008	6.6663	0.195	1.2999
13	50.0924	0.4179	-4.5601	-0.7117	2.2	-1.5658
14	112.4129	7.128	-26.4571	8.8663	2.75	14.3822
15	59.7172	13.3487	-27.9889	8.8663	1.45	12.8561
21	186.2913	1.5303	-10.8257	11.4137	2.75	31.3878
22	98.2263	5.8894	-23.7806	11.4137	1.45	16.5499
31	37.1998	2.6175	7.7331	-6.6663	3.805	-25.3651
32	39.0329	25.6957	30.9425	-6.6663	3.9925	-26.615
33	26.5363	44.4486	34.1342	-7.2808	2.15	-15.6537
41	112.4129	1.5303	8.4095	-8.8663	2.75	-14.3822
42	59.7172	5.8894	18.4729	-8.8663	1.45	-12.8561

Associated Printers, Inc. • (708) 367-1009

In that mass-production process, some operations are purely “**node-driven**”, some are purely “**segment-driven**”, while some others are mixed. In light of the sheer amount of data involved at various stages (steps), it would be impossible to avoid but to present the transition data by means of tabulation. Based on the “look” from each Calc page, we shouldn’t be surprised that we must undergo many cycles of data “storing”, “retrieving” and “manipulating” before arriving at  $C_w$ .

As some of you who grew comfortable with spreadsheets would have guessed: this “homework problem” fits perfectly for a spreadsheet-like application.

Let us also clarify: the term “**spreadsheet-like**” can be “**database-like**” as well. Let’s not debate on how easy or otherwise, to implement applications using either type of **automation tool**. Nevertheless it pays to understand the process at hand first and then plan out the overall solution strategy in terms of what either tool can offer. For that we are not going to be shy from borrowing a mixed bag of spreadsheet and/or **database terms** to put across my points from time to time (as we already have).

## Data Manipulation

After we completed the input for both tables, our very next chore would be to retrieve the coordinates of starting (From) and ending (To) nodes for each segment and get ready to locate the centroid of our profile.

Since all the **source** coordinates are stored in “Node table”, the chore can easily be accomplished by using “LOOKUP” Function or an Inner-joint **Query** in that the “From Node” (or the “To Node”) of

a segment in the “Segment table” **matches** the Node ID in the “Node table”. The result from this practice are shown under the last four (4) columns in the Segment tabulation as  $[x_i]$ ,  $[y_i]$ ,  $[x_j]$  and  $[y_j]$  on Calc page 2. Implicitly we assigned “i” as the index for “From node” and “j” as the index for “To” node.

The “Segment tabulation” now combines both the “original input columns” and the “LOOKUPed coordinate columns”. This would constitute a newly expanded “database **View**” forming the basis for further data process. By way of this practice we exhibited the simplest form of “**Data Merging**” or “**Query**” after its action. The “lookup” feature is a standard protocol in **SELECT Query**. The columns of data displayed in the “View” are the result of “what we pick is what we get”.

The “LOOKUP” Function (or feature) is among one of many predefined library Functions. These Functions are inherent to the software, available for us to “Call” up

to “look something up” or to perform numerical operations like taking the Square Root of a number. Besides that, users like us can also define our own Functions explicitly (or implicitly depending on how we qualify them) for our own purpose.

“Simple” Functions or “program procedures” defined by the user often involve only a handful of data operators and operands placed in one or several fields (cells). They are usually entered directly as an “in-line statement” into a database Query field or a Cell in the spreadsheet. It is very convenient to use them for purposes as simple as adding or subtracting the value from one field (cell) to the value in other fields (cells). Here are a couple of examples:

- (a) using expression  $[L_x] = [x_j] - [x_i]$  to calculate the length of a segment projected onto the x-axis then assign the result to  $[L_x]$ ,

**SD** Structural Design Corporation

Client: \_\_\_\_\_

Project: SDC STANDARD

Calcs. For: WARPING CONSTANT OF GIRDER

W21X55 + C8X11.5

Prepared by: JOHN FONG

Reviewed by: \_\_\_\_\_

Calc. No. \_\_\_\_\_

Rev. \_\_\_\_\_ Date \_\_\_\_\_

Page 6 of 16

Date: 7-3-10

**7. PRINCIPAL MOMENT OF INERTIA**

Σ USER AXIS MOMENT OF INERTIA:

$$\begin{cases} I_{xx} = 1314.7098 \\ I_{yy} = 132.438 \\ I_{xy} = 86.079 \end{cases}$$

LET  $z = [(I_{xx} - I_{yy})^2 / 4 + I_{xy}^2]^{1/2} = 597.3703$

$\alpha = \frac{1}{2} \tan^{-1} \left( \frac{2 I_{xy}}{I_{yy} - I_{xx}} \right) = -20.723^\circ = -4.1425^\circ$

PRINCIPAL MOMENT OF INERTIA:

$$\begin{cases} I_{x'} = (I_{xx} + I_{yy}) / 2 + z = 1320.9442 \\ I_{y'} = (I_{xx} + I_{yy}) / 2 - z = 126.2036 \end{cases}$$

**8. NODAL COORDINATE TRANSFORMATION PER PRINCIPAL AXES**

FROM USER AXIS  $\begin{Bmatrix} x \\ y \end{Bmatrix}$   $\xrightarrow[\text{(TRANSLATION ONLY)}]{\text{TRANSFORM}}$  CENTROIDAL AXIS  $\begin{cases} x_c = x - C_x \\ y_c = y - C_y \end{cases}$

FROM USER AXIS  $\begin{Bmatrix} x \\ y \end{Bmatrix}$   $\xrightarrow[\text{(TRANSLATION + ROTATION)}]{\text{TRANSFORM}}$  PRINCIPAL AXIS  $\begin{cases} x' = y_c \sin \alpha + x_c \cos \alpha \\ y' = y_c \cos \alpha - x_c \sin \alpha \end{cases}$

SEGMENT ID	NODE		TRANSFORMED COORDINATES				SEGMENT CENTROID	
	FROM <sub>i</sub>	TO <sub>j</sub>	$x'_i$	$y'_i$	$x'_j$	$y'_j$	$x'_c$	$y'_c$
4	4	5	-4.0744	-11.7387	-2.6282	-11.634	-3.3513	-11.6864
5	5	6	-2.6282	-11.634	0.1146	-11.4353	-1.2568	-11.5347
6	6	7	0.1146	-11.4353	-0.2422	-6.5082	-0.0638	-6.9718
21	6	21	0.1146	-11.4353	1.8574	-11.2367	1.486	-11.336
7	7	8	-0.2422	-6.5082	-0.7912	1.0719	-0.5167	-2.7182
8	8	9	-0.7912	1.0719	-1.0361	4.453	-0.9137	2.7625
9	9	10	-1.0361	4.453	-0.8416	4.4671	-0.9389	4.4601
10	10	11	-0.8416	4.4671	-0.9055	5.3498	-0.8736	4.9085
11	11	12	-0.9055	5.3498	-0.9969	6.6115	-0.9512	5.9807

Associated Printers, Inc. • (708) 367-1009



- (b) using  $[L] = \text{SQR}([L_x]^2 + [L_y]^2)$  to calculate the true length of a segment then assign the result to  $[L]$ .

In more complex situations, these functions can be custom-developed full-blown application Functions stored in separate **program modules**. By design, a Query can manipulate its operand data residing in:

- (a) a single table or several discrete tables, and/or involving
- (b) other sibling Queries (participating as sub-queries) through data links (or none at all)

to create a new database “**View**”.

Data tucked behind each View is called “**Recordset**”. In relational database applications, the data-linking relationship(s) can:

- (a) be predefined as in this case that the “Node ID” from “Node table” is linked through an “**one-to-many**” relationship to both the “From Node” and the “To Node” in “Segment table” or,
- (b) be established on the fly (known as late-binding technique) for application data lacking preset relationships.

Also by specifying data domain criteria, Queries can be controlled “by design” to manipulate either the entire collection or a handful of data records associated with the active Recordset domain. For different purposes the resulting database “**View**” from a Query can contain:

- (a) a multi-rowed tabulation collecting all the domain records to become a “**Full-Domain Tabulation View**” or
- (b) just one single “summary line” displaying the return value from application of “**Domain Aggregate Functions**” to the entire Recordset becoming a “**Summary line View**”.

Among others, “Maximum”, “Minimum”, “Average” and “Total” (or “Summation”) are some of the most common uses of **domain aggregate functions** in engineering applications.

Both types of data “View” are essential and equally important to our calculation. Notice that the “Full-Domain Tabulation View” appeared more often than “Summary line View” in this Calc. Let us review just a few Calculation statements near the top of Calc page 4:

- (a) terms  $[SA]$ ,  $[SAX]$  and  $[SAY]$  are the results from applying “Summation” (Domain Aggregate Function) to terms  $[A]$ ,  $[AX]$  and  $[AY]$ , respectively from Segment Properties’ Recordset already tabulated on Calc page 3.
- (b) The “Summary line View” had been subtly hidden behind the scene in the Calc. Instead of tabulating them we had expressed the results as if we manually calculated the centroid coordinates  $[cx]$  and  $[cy]$ .

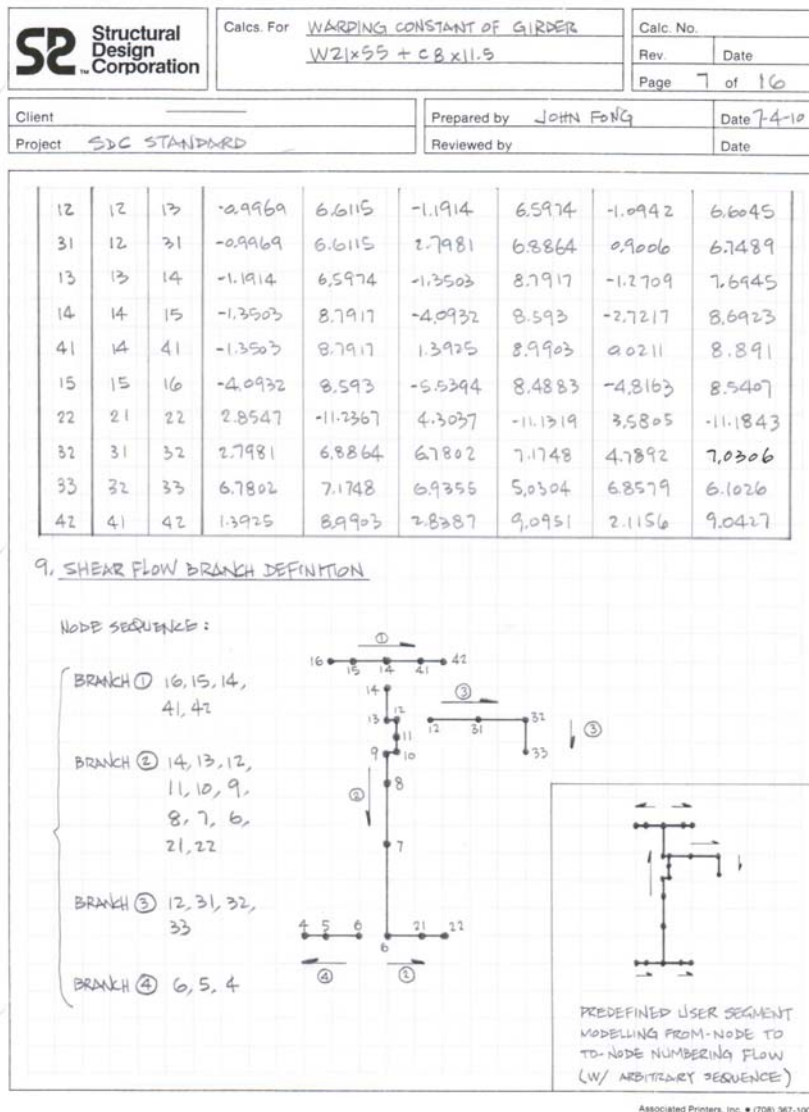
## Principal Axes

Once the coordinates of centroid are located, the profile moment of inertia with respect to (w.r.t.) the user axes,  $[I_{xx}]$ ,  $[I_{yy}]$  and  $[I_{xy}]$ , can be calculated. From that the orientation angle  $[\alpha]$  of principal axes w.r.t. the user axes and the moment of inertia w.r.t. the principal axes,  $[I_x']$  and  $[I_y']$ , would follow.

Before moving on to shear center-based properties we need to finish up with three (3) more centroid-based properties:

- (a) the distance  $[p]$  for each segment measured from the “centroid” orthogonal to the “segment tangent line”. Tangent line is formed by connecting the “From Node” to “To Node” for the respective segment,

- (b) the Double Sectorial Area [pds] w.r.t. centroid for each segment, (see algebraic formulation on Calc page 4) and
- (c) the nodal coordinates, transformed from user-axes based system into principal axes based system:
- through translation only:  $[x_i - cx]$ ,  $[y_i - cy]$ ,  $[x_j - cx]$  and  $[y_j - cy]$  for shear center-based properties,
  - through combination of translation and rotation,  $[x'_i]$ ,  $[y'_i]$ ,  $[x'_j]$  and  $[y'_j]$  for flexural-based properties that are important for unsymmetrical bending.



To relish the power of Query, all the centroid-based properties covered so far (including the last three properties), as shown on Calc pages 2 through 7, can be “calculated” by a series of SELECT Query working in chain reaction. The results from a leading Query can be “Looked Up” by a subsequent Query and then be processed further all the way from original “Node” and “Segment” table data down to the final “nodal coordinate transformation”. Intermediate results from any Query in the chain can be brought to “View” at any instance.

Source data records stored in a “Table” is treated the same as that in a “Query”. In Data Access’ term,

records from both sources are collectively called “Recordset”. They even share the same “View” or “Look” when “Previewed” or “Opened” within the database desktop. The difference is that the data in a “Tables” are “Static” whereas data in a “SELECT Query” are “Dynamic”. Being dynamic means any changes made in the source Table (or Query) content would trigger an instant (cascading)

update(s). The update applies to Recordset behind all the SELECT Queries that have association with the source data. "Updates" are done automatically regardless if these Queries are dormant or "Opened" in user/desktop views or not.

"Dynamic" behavior should not affect the Query setup for our calculation at hand. We point this out as a caution on **"Dynamic Recordsets"**: if you make a "Function call" from within a chained Query that manipulates data derived from another preceding (feeding) Query and that piece of data happen(s) to be **"Null"** then you should **watch out for unexpected results**.


## Shear Flow Scheme

Without knowing their intensity and how the shear stresses were actually distributed over the profile, "shear flow" is at best a graphical prediction of stress vectors. These vectors represent the user's view (or illustration) of "shear stress flow orientation" traversing each and every segment for a given profile. In order to work out the "proper" location of shear center, this is the very first step no matter how simple (straightforward) or how complex the segment arrangement may be.

Channels, arcs, zeers or any single-line linkages are example profiles having **straightforward** "segment arrangement". By "straightforward"

we mean the most basic profile shape that always conforms to singular continuity. Their middle-line models can be drawn using one **"single stroke"**. If one stroke represents one "shear flow branch", then these shapes would have only one single flow branch. Knowing how this works we can "define" the shear flow pattern for any open sections. Here in the homework problem we are dealing with a profile having **not-so-straightforward** segment arrangement. Its shear flow pattern has to be drawn with multiple strokes or **multiple flow branches**.

As depicted in Calc step 9, our model takes **four (4) strokes** to complete as we have chosen to define our "four-branched" sample. **Are there any other ways to define that?** The answer is



Calc. For WARPING CONSTANT OF GIRDER  
W21x55 + C8x11.5

Calc. No. \_\_\_\_\_  
 Rev. \_\_\_\_\_ Date \_\_\_\_\_  
 Page 8 of 16

Client \_\_\_\_\_  
 Project SDC STANDARD

Prepared by JOHN FONG Date 7-4-10  
 Reviewed by \_\_\_\_\_ Date \_\_\_\_\_

### 10. INTEGRATION OF DOUBLE SECTORIAL AREA W.R.T. CENTROID (W)

BRANCH ID	N O D E	INCIDENCE	TO	FROM	KNOWN Pds	Pds FOR INTEGRATION	RESULTING NODAL $\omega = \int Pds$	BRANCH INITIAL $\int Pds$
1	1	10	15	-1	12.8561	-12.8561	0	
	2	15	14	-1	24.3822	-24.3822	12.8561	
	3	14	41	1	-24.3822	-24.3822	-37.2383	
	4	41	42	1	-12.8561	-12.8561	-61.6204	
	5	42	-	-	-	-	-74.4765	
2	1	14	13	-1	-1.5658	1.5658	-37.2383	-37.2383
	2	13	12	-1	1.2999	-1.2999	-35.6725	FROM NODE 14 BRANCH 1
	3	12	11	-1	-0.6537	0.6537	-36.9724	
	4	11	10	-1	-0.4573	0.4573	-36.3187	
	5	10	9	-1	-0.8807	0.8807	-35.8614	
	6	9	8	-1	-2.4128	2.4128	-34.9807	
	7	8	7	-1	-5.4091	5.4091	-32.568	
	8	7	6	-1	-3.5159	3.5159	-27.1589	
	9	6	21	1	31.3878	31.3878	-23.6429	
	10	21	22	1	16.5499	16.5499	7.7449	
3	1	12	31	1	-15.3651	-25.3651	-36.9724	-36.9724
	2	31	32	1	-26.615	-26.615	-62.3375	FROM NODE 12 BRANCH 2
	3	32	33	1	-15.6537	-15.6537	-88.9525	
	4	33	-	-	-	-	-104.6061	
4	1	6	5	-1	31.3878	-31.3878	-23.6429	-23.6429
	2	5	4	-1	16.5499	-16.5499	-55.0307	FROM NODE 6 BRANCH 2
	3	4	-	-	-	-	-71.5807	

Associated Printers, Inc. • (708) 387-1009

“Yes” provided that we follow a certain “scheme”. Let us draw the strokes in “unidirectional manner” using my Scheme for this example:

- starting arbitrarily the 1<sup>st</sup> flow branch from any terminal node, say node [16];
- moving along segments at will through as many nodes [15, 14, 41,...] as we encounter in between;
- finishing the 1<sup>st</sup> flow branch upon reaching another terminal node [42];
- diverging the 2<sup>nd</sup> branch off any interfacing node, say node [14] from the 1<sup>st</sup> branch then completing the 2<sup>nd</sup> branch as in steps (b) and (c) [14, 13, 12,...21, 22];
- diverging subsequent branches off from a node on any established branches the same way until depleting all segments without duplicating usage of any one of them.

Imagine if it were a “tree structure”, we may visualize the 1<sup>st</sup> branch as “trunk” and all the other offshoot branches as “twigs”. Here is another example following the same “scheme” but taking different routes that we end up with:

- branch 1 as: from node 16, 15, 14, 13, 12, 31, 32 to 33;
- branch 2 as: from node 12, 11, 10, 9, 8, 7, 6, 5 then 4;

	Calcs. For <u>WARPING CONSTANT OF GIRDER</u>				Calc. No. _____	
	<u>W21x55 + CBx11.5</u>				Rev. _____	Date _____
					Page <u>9</u> of <u>16</u>	
Client _____			Prepared by <u>JOHN FONG</u>			Date <u>7-4-10</u>
Project <u>SDC STANDARD</u>			Reviewed by _____			Date _____

11. SUMMARY OF SEGMENTAL PARAMETERS

SEGMENT ID	FROM NODE I				TO NODE J			
	NODE ID	$\omega_i$	PRINCIPAL AXES COORDINATES		NODE ID	$\omega_j$	PRINCIPAL AXES COORDINATES	
			$X_i - C_x$	$Y_i - C_y$			$X_j - C_x$	$Y_j - C_y$
4	4	-71.587	-4.9117	-11.4137	5	-55.0307	-3.4617	-11.4137
5	5	-55.0307	-3.4617	-11.4137	6	-23.6429	-0.7117	-11.4137
6	6	-23.6429	-0.7117	-11.4137	7	-27.1589	-0.7117	-6.4737
7	7	-27.1589	-0.7117	-6.4737	8	-32.568	-0.7117	1.1263
8	8	-32.568	-0.7117	1.1263	9	-34.9807	-0.7117	4.5163
9	9	-34.9807	-0.7117	4.5163	10	-35.8614	-0.5167	4.5163
10	10	-35.8614	-0.5167	4.5163	11	-36.3187	-0.5167	5.4013
11	11	-36.3187	-0.5167	5.4013	12	-36.9724	-0.5167	6.6663
12	12	-36.9724	-0.5167	6.6663	13	-35.6725	-0.7117	6.6663
13	13	-35.6725	-0.7117	6.6663	14	-37.1383	-0.7117	8.8663
14	14	-37.1383	-0.7117	8.8663	15	-12.8561	-3.4617	8.8663
15	15	-12.8561	-3.4617	8.8663	16	0	-4.9117	8.8663
21	6	-23.6429	-0.7117	-11.4137	21	7.7449	2.0383	-11.4137
22	21	7.7449	2.0383	-11.4137	22	24.2948	3.4883	-11.4137
31	12	-36.9724	-0.5167	6.6663	31	-62.3375	3.2883	6.6663
32	31	-62.3375	3.2883	6.6663	32	-88.9525	7.2808	6.6663
33	32	-88.9525	7.2808	6.6663	33	-104.6061	7.2808	4.5163
41	14	-37.1383	-0.7117	8.8663	41	-61.6204	2.0383	8.8663
42	41	-61.6204	2.0383	8.8663	42	-74.4765	3.4883	8.8663

12. SEGMENTAL WARPING PRODUCT OF INERTIA

Associated Printers, Inc. • (708) 387-1009



(c) branch 3 as: from node 14, 41 to 42; and

(d) branch 4 as: from 6, 21 to 22, or any other arrangement(s).

Notice that the flow “direction” along these branches defined in both examples does not conform to any explicit rule in sign conventions. In other words, both examples employed no assumptions on the **generic positive** (or negative) flow senses. Neither has it any bearing on how the segment nodal incidences were assigned in the **original Segment input** (see inset at lower right-hand corner on Calc page 7).


Therefore it really doesn’t matter how “wrong” our prediction on the stress flow “senses” that may be. Consequently all the “wrongs”, if ever true, would be “corrected automatically” after all the numerical

integrations are completed. This “auto-correction” would work for any profile geometry so long as their branch assignments and the associated “**node sequences**” were set up **consistently** using this “scheme”.

Once they are finalized, it is important to store the shear flow branch data. Because we have to refer to them for at least three (3) times from here on, it’s preferable to save them in a **table** rather than plain **program variables (arrays)**. This important table we may name it “ShearFlowID”, “ShearFlowBranches”, or simply “**Branches**”.

The “**Branches**” table should have at least four (4) fields to store:

(a) Shear Flow Branch ID,



Calcs. For WARPING CONSTANT OF GIRDER

W21x55 + C8x11.5

Calc. No. \_\_\_\_\_

Rev. \_\_\_\_\_ Date \_\_\_\_\_

Page 10 of 16

Client \_\_\_\_\_

Project SDC STANDARD

Prepared by JOHN FONIG

Reviewed by \_\_\_\_\_ Date \_\_\_\_\_

$I_{wx} = \text{WARPING PRODUCT OF INERTIA ABOUT PRINCIPAL } x_c$

$$= \frac{1}{3} \sum_{i=0}^n (\omega_i x_i + \omega_j x_j) b_{ij} t_{ij} + \frac{1}{6} \sum_{i=0}^n (\omega_i x_i + \omega_j x_j) b_{ij} t_{ij}$$

$x_{PA}$  $x_{PB}$

$I_{wy} = \text{W.P. I. ABOUT PRINCIPAL } y_c$

$$= \frac{1}{3} \sum_{i=0}^n (\omega_i y_i + \omega_j y_j) b_{ij} t_{ij} + \frac{1}{6} \sum_{i=0}^n (\omega_i y_i + \omega_j y_j) b_{ij} t_{ij}$$

$y_{PA}$  $y_{PB}$

SEGMENT ID	A = $b_{ij} t_{ij}$	INTEGRATION ABOUT $x_c$			INTEGRATION ABOUT $y_c$		
		$x_{PA}$	$x_{PB}$	$\frac{x_{PA}}{3} + \frac{x_{PB}}{6}$	$y_{PA}$	$y_{PB}$	$\frac{y_{PA}}{3} + \frac{y_{PB}}{6}$
4	0.754	408.7329	390.6388	201.3508	1089.6136	1089.6136	544.8068
5	1.43	296.48	171.0475	127.6679	1284.0852	1284.0852	642.0426
6	1.8525	66.9808	66.9808	33.4904	825.6115	857.787	418.1683
7	2.85	121.1512	121.1512	60.5756	396.5489	513.7105	217.8014
8	1.2713	61.1169	61.1169	30.5585	-247.4635	-237.0657	-121.9988
9	0.0002	9.0085	0.0085	0.0042	-0.0624	-0.0624	-0.0312
10	0.677	25.2513	25.2513	12.6256	-242.4601	-242.1861	-121.1844
11	0.9677	36.6491	36.6491	18.3246	-428.3674	-427.5472	-214.0403
12	0.0002	0.0087	0.0087	0.0043	-0.0944	-0.0944	-0.0472
13	0.825	42.8113	42.8113	21.4057	-468.5715	-465.7296	-233.8121
14	1.43	101.5409	197.4239	66.7509	-635.1328	-635.1328	-317.5664
15	0.754	32.5561	47.6117	19.1207	-85.9447	-85.9447	-42.9724
21	1.43	46.6373	-76.7953	2.7465	259.4829	259.4829	127.7414
22	0.754	75.8019	57.7079	34.8853	-275.7324	-275.7324	-137.8662
31	0.8371	-155.5985	-74.8065	-64.3339	-554.1805	-554.1805	-277.0903
32	0.8784	-748.9034	-655.5696	-358.8961	-885.848	-885.848	-442.924
33	0.8385	-1181.661	-1181.661	-590.8305	-893.3435	-921.5635	-451.3751
41	1.43	-141.7071	-45.8241	-54.873	-1253.4037	-1253.4037	-626.7018
42	0.754	-290.5868	-276.5313	-142.9508	-909.8289	-909.8289	-454.9144

Associated Printers, Inc. • (708) 367-1008

(b) Nodal  
Sequence  
(Serial) ID,

(c) Node ID and

(d) Incidence  
Segment ID.

Much have we tried, there seems to be no **“practical way”** to derive the “shear flow definition data” using database “Queries” alone. In two ways we can prepare the records for “Branches” table by:

(a) filling in manually after defining shear flow branches on a sketch similar to that as shown in step 9 on Calc page 8, or

(b) filling in automatically by a custom-made **Program** to carry out the “scheme” (This is an excellent challenge for you engineering programmers).

	Calcs. For <u>WARPING CONSTANT OF GIRDER</u> <u>W21x55 + CBx11.5</u>		Calc. No. _____
	Rev. _____ Date _____		Page <u>11</u> of <u>16</u>
	Client _____		Prepared by <u>JOHN FONG</u> Date <u>7-4-10</u>
Project <u>SDC STANDARD</u>		Reviewed by _____ Date _____	

$I_{wx} = \sum \left( \frac{xPA}{3} + \frac{XPB}{6} \right) = -582.3733$   
 $I_{wy} = \sum \left( \frac{YPA}{3} + \frac{YPB}{6} \right) = -1489.9641$

13. LOCATION OF SHEAR CENTER (S.C.)

Let  $D_m = I_{xy}^2 - I_{xx} I_{yy}$   
 $= (86.079)^2 - (1314.7098)(132.438) = -166707.9262$

$x_{ref} = (I_{xy} I_{wx} - I_{yy} I_{wy}) / D_m$   
 $= [(86.079)(-582.3733) - (132.438)(-1489.9641)] / D_m$   
 $= 147197.7326 / (-166707.9262) = -0.883$

$y_{ref} = (I_{xx} I_{wy} - I_{xy} I_{wx}) / D_m$   
 $= [(1314.7098)(-1489.9641) - (86.079)(-582.3733)] / D_m$   
 $= (-637397.2861) / (-166707.9262) = 3.8234$

C.G.  $\begin{cases} C_x = 0.7117 \\ C_y = 11.6737 \end{cases}$

S.C. W.R.T. USER AXES  $\begin{cases} x_o = x_{ref} + C_x = -0.1712 \\ y_o = y_{ref} + C_y = 15.4972 \end{cases}$

14. INTEGRATION OF DOUBLE SECTORIAL AREA W.R.T. SHEAR CENTER ( $\omega_o$ )


$\rho_o$  = NORMAL DISTANCE FROM SHEAR CENTER TO SEGMENT TANGENT  
 $= -(L_y x_o - L_x y_o - x_i y_i + x_j y_j) / L$   
 $\omega_o$  = DOUBLE SECTORIAL AREA W.R.T. S.C.  
 $= \int \rho_o ds$

\*  $I_{xx}, I_{xy}, I_{yy}$  ARE USER AXIS BASED MOMENT OF INERTIA

Associated Printers, Inc. • (708) 367-1009

Some of the formulas in calculating **shear center-based properties** appear very intimidating when expressed in algebraic terms. But to carry out the calculation, it is actually a very clear-cut process so long as we don't misuse or mix up the terms in each step. Anyone can do it simply by following a set procedure. Numerically it is always forgiving even if we plugged in **flawed** or **wrong shear flow data** to the formulas. It would still lead us to the “false location of shear center” and wrong value of  $C_w$  with no forewarning. On the whole it is never forgiving for any negligence leading to “garbage in garbage out” results. To be “wrong” we prefer it be wrong “physically” rather than “logically”. To make sure we do it “**logically correct**” in defining shear flow data, it is essential to verify the followings:

(a) complete the “trunk” definition first before moving on to the “twigs”



Calcs. For: WARPING CONSTANT OF GIRDER

W21x55 + CBx11.5

Calc. No. \_\_\_\_\_

Rev. \_\_\_\_\_ Date \_\_\_\_\_

Page 12 of 16

Client: \_\_\_\_\_

Project: SDC STANDARD

Prepared by: JOHN FONG

Reviewed by: \_\_\_\_\_

Date: 7-5-10

Date: \_\_\_\_\_

PROCESS SIMILAR TO INTEGRATION OF DOUBLE SECTORIAL AREA W.R.T. CENTROID

BRANCH ID	NODE SEQUENCE	NODE ID	INCIDENCE	S E G M E N T		ds (1)	P <sub>0</sub> ds	RESULTING NODAL ω <sub>0</sub> = ∫ P <sub>0</sub> ds	BRANCH INITIAL P <sub>0</sub> ds
				FROM TO NODE	P <sub>0</sub>				
1	1	16	15	-1	5.0428	1.45	-7.3121	-21.1798 FROM NODE 14 BRANCH 1	
	2	15	14	-1	5.0428	2.75	-13.8677		
	3	14	41	1	-5.0428	2.75	-13.8677		
	4	41	42	1	-5.0428	1.45	-7.3121		
	5	42	-	-	-	-	-42.3596		
2	1	14	13	-1	0.1712	2.2	-0.3767	-22.1109 FROM NODE 12 BRANCH 2	
	2	13	12	-1	2.8428	0.195	-0.5543		
	3	12	11	-1	0.3662	1.265	-0.4633		
	4	11	10	-1	0.3662	0.885	-0.3241		
	5	10	9	-1	-0.6928	0.195	0.1351		
	6	9	8	-1	0.1712	3.39	-0.5805		
	7	8	7	-1	0.1712	7.6	-1.3014		
	8	7	6	-1	0.1712	4.94	-0.8459		
	9	6	21	1	15.2372	2.75	41.9023		
	10	21	22	1	15.2372	1.45	22.0939		
	11	22	-	-	-	-	38.5051		
3	1	12	31	1	-2.8428	3.805	-10.8169	-25.4911 FROM NODE 6 BRANCH 2	
	2	31	32	1	-2.8428	3.9925	-11.3499		
	3	32	33	1	-8.1637	2.15	-17.552		
	4	33	-	-	-	-	-61.8298		
4	1	6	5	-1	15.2372	2.75	41.9023		
	2	5	4	-1	15.2372	1.45	22.0939		
	3	4	-	-	-	-	-89.4873		

Associated Printers, Inc. • (708) 367-1009

- (b) make sure each twig is diverging off a node on an established branches (i.e. twigs do not cross over the trunk or over each other);
- (c) without duplicating usage of any segment, make sure that all the nodes and segments have participated in the flow branch definition;
- (d) the total number of records in the "Branches" table equals to the sum of "Number of Segment" plus "Number of Branches";
- (e) the flow "sense" from node to node should be unidirectional

with no reversal within the same "Branch."

After practicing the shear flow scheme a few times you may have notice a few interesting facts that it doesn't matter

- (a) where each node is located,
- (b) how thick each segment may be,
- (c) how many strokes it took to draw up the profile and
- (d) how we sequence the nodes.

We point this out to prove a point that shear flow scheme is a "pure logic process". Our "scheme" utilizes only three (3) items: Segment ID, Segment From-node and To-node IDs from "Segment" table without incurring any reference to "Node table" at all. The logic sounds simple but could be a real challenge to automate the shear flow assignment process together with the ensuing procedures for numerical integrations toward  $C_w$ .

### Double Sectorial Area w.r.t. Centroid (ω)

“Summation” is the simplest form of numerical integration. Recall that we first applied “Summation” method to locate the profile centroid and then obtain the user-axes based moment of inertia. This method unfortunately does not work for integration of **Double Sectoria Area (DSA)** and other “shear flow-related integrations”. Instead we must proceed in a step-by step (node-by-node) manner. Here is how it works in a nutshell, the “**succeeding value**” of integration at any intermediate step is dependent on:

- (a) its own attributes and
- (b) the “**preceding value**” determined from a previous step.

This suggests that we must progress the nodal data integration in an ascending (or descending) order according to the shear flow logic.

To carry out the  $\omega = \int pds$  integration for **DSA**, w.r.t. centroid, let us examine how it’s done (per step 10 on Calc page 8):

- (a) We begin with completion of the first four (4) columns by simply repeating (copying) the **ID/serial data** from the “Branches table”.
- (b) focusing on the **5<sup>th</sup>** column tabulation: for a given node ID appearing in the **3<sup>rd</sup>** column and its corresponding segment ID in **4<sup>th</sup>** column, with additional reference to the “Segment” table, based on its input “From-Node and To-Node” logic pattern, define the **Segment-Node-Incidence Sign convention (SNIS)**. Fill in the **SNIS** under the 5<sup>th</sup> column. Apply the sign convention positive (+1) if current incidence node is a From-node, negative (-1) if incidence node is a To-node.

The key is to apply the “Sign Convention Strategy” **consistently** to all nodes. Remember that we made a statement: “it really **doesn’t matter** how wrong our prediction on the stress flow senses that may be”. Well, now is the time to rectify through **properly** assigned sign conventions. **SNIS** would **automatically** redeem for us if we ever were “wrong” to begin with. Because it takes a lot of “back and forth” looking around between the current Calc tabulation and the Segment table, this step turns out to be the **most crucial step** in all.

- (c) look up the **pds** value from Calc step 6 for segment identified in the 4<sup>th</sup> column, then tabulate the resulting **pds** to the cell under [**KNOWN pds**] column;
- (d) calculate the product of [**SNIS**] \* [**KNOWN pds**] for the current segment then place the result under column “**pds FOR INTEGRATION**” value (**SEGV**);
- (e) the “**RESULTING NODAL  $\omega = \int pds$** ” value (**RNV**) can be interpreted as the instantaneous result of “ $\omega = \int pds$  integration” we are interested in, at the current node, on a given branch. It is the “Sum” of “**RNV**” already known at the preceding node and the “**SEGV**” of the segment where the current node has incidence with.

Data entered from step (b) through step (d) were only the “warm-ups” for each individual node/segment. The real deal is the “process outlined in step (e)” that does the actual **integration towards “DSA for the entire profile”**. Step (e) should be repeated “in succession” conforming to the “ascending” serial order assigned for each node on the given branch until finishing with the branch.

Whenever a new branch is **initiated** at a certain Node, the initial **RNV**:

- (a) is either zero (0) (only once when integration process has just begun), or
- (b) must be **LOOKed UP** from another node having the same ID existed on a pre-processed branch appearing under “**BRANCH INITIAL  $\int pds$** ”.

This integration process is much easier “done” than “said”. The “step/process” is not complete unless we carried out fully from branch 1, node serial 1 all the way through the last node on the last branch. A sample worked-out on the “**Node-driven**” data flow for branch #1 can be traced by following the dotted arrow as shown on Calc page 8.


## Warping Moment of Inertia



“Warping Moment of Inertia” about Principal x’ and y’ Axes (**WMOIPA**) are designated as **I<sub>ωx</sub>** and

**I<sub>ωy</sub>**, respectively.

They carry a dimension unit of (Length)<sup>5</sup> versus the (Length)<sup>4</sup> for Flexural Moment of Inertia.

	Calcs. For	WARPING CONSTANT OF GIRDER	Calc. No.	
		W21x55 + C8x11.5	Rev.	Date
			Page 13 of 16	
Client			Prepared by	JOHN FONG
Project	SDC STANDARD		Reviewed by	
			Date	7-5-10

15. NORMALIZED UNIT WARPING ( $\omega_n$ )

$$\omega_n = \frac{1}{\sum A} \int_0^L \omega_0 t ds - \omega_0 = \frac{1}{\sum A} \left[ \sum_{\text{SEGMENTS}} \frac{\omega_{0i} + \omega_{0j}}{2} (t ds) \right] - \omega_0 = C - \omega_0$$

SEGMENT ID	NODE		$\omega_0$		$t ds$ (A)	$(\omega_i + \omega_j)(t ds)$	NODE ID	$\omega_0$	$\omega_n = C - \omega_0$
	FROM	TO	$\omega_{0i}$	$\omega_{0j}$					
10	10	11	-22.8983	-22.5742	0.677	-30.786	10	-22.8983	-2.6451
11	11	12	-22.5742	-22.1109	0.9677	-43.1429	11	-22.5742	-2.9692
12	12	13	-22.1109	-21.5565	0.0002	-0.0085	12	-22.1109	-3.4325
13	13	14	-21.5565	-21.1798	0.825	-35.2575	13	-21.5565	-3.9868
14	14	15	-21.1798	-7.3121	1.43	-40.7434	14	-21.1798	-4.3656
15	15	16	-7.3121	0	0.764	-5.5133	15	-7.3121	-18.7313
21	6	21	-25.4911	16.4112	1.43	-12.9843	16	0	-25.5434
22	21	22	16.4112	38.5051	0.754	41.4069	21	16.4112	-4.19545
31	12	31	-22.1109	-32.9278	0.8371	-46.0729	22	38.5051	-64.0485
32	31	32	-32.9278	-44.2777	0.8783	-67.8135	31	-32.9278	7.3844
33	32	33	-44.2777	-61.8298	0.8385	-88.9711	32	-44.2777	18.7344
4	4	5	-89.4873	-67.3934	0.754	-118.288	33	-61.8298	36.2864
41	14	41	-21.1798	-35.0476	1.43	-80.4051	4	-89.4873	63.9439
42	41	42	-35.0476	-42.3596	0.754	-58.365	41	-35.0476	9.5042
5	5	6	-67.3934	-25.4911	1.43	-132.8248	42	-42.3596	16.8163
6	6	7	-25.4911	-24.6542	1.8525	-92.8774	5	-67.3934	41.85
7	7	8	-24.6542	-23.3437	2.85	-136.7683	6	-25.4911	-0.0523
8	8	9	-23.3437	-22.7632	1.1713	-58.6135	7	-24.6542	-0.8982
9	9	10	-22.7632	-22.8983	0.0002	-0.0089	8	-23.3437	-2.1996
C = $\frac{-1008.1375}{2 \times 19.7338} = -25.5434$					19.7338	-1008.1375	9	-22.7632	-2.7802
					$\Sigma$	$\Sigma$			

Associated Printers, Inc. • (708) 367-1009

Calc pages 10 and 11.

## Shear Center

On occasions if we deal with members of doubly- or mono- symmetric profiles time and again, we could get so used to and feel so comfortable with the fact that “the orientation of user-chosen-axes for middle-line profile model is same as that for the principal axes”, for which both the orientation angle  $[\alpha]$  and the product moment of inertia  $[I_{xy}] = 0$ . This special condition due to axis symmetry leads to great degrees of simplification in our practices. It simplifies every formula used to calculate profile nodal stresses, material allowable stresses and/or section properties including formulas for the location of shear center (read on for more details on this).

Let’s review, among all the transition parameters and section properties we have covered so far, and see what we need in order to locate the shear center:

(a) “Flexural Moment of Inertia” w.r.t. User-Axes;  $I_{xx}$ ,  $I_{yy}$  and  $I_{xy}$

(b) “Warping Moment of Inertial” w.r.t. Principal Axes;  $I_{\omega x}$  and  $I_{\omega y}$  and

(c) coordinates of profile centroid;  $c_x$  and  $c_y$ .

Let’s also define a few shear center-related **unknowns**:

(a)  $X_{ref}$  and  $Y_{ref}$  to be the relative displacements (translation distances) from the profile centroid to the shear center and

(b)  $x_o$  and  $y_o$  to be the shear center coordinates based on user-axes system.

Fitting different categories of profile geometry, there are three (3) cases of shear center formulas for  $X_{ref}$  and  $Y_{ref}$  we need to get familiar with.

**Case 1:** the simplest case applicable to the most obvious member category with doubly- symmetric profiles: **the shear center always coincides with the centroid** rendering both  $X_{ref}$  and  $Y_{ref} = 0$ , and therefore  $x_o = c_x$  and  $y_o = c_y$ .

**Case 2:** the most generic case applicable to all other profile geometries (with or without any axis symmetry), we may find them from many textbooks the expressions for calculating the coordinates of shear center as:

$$X_{ref} = (I_{xy} * I_{\omega x} - I_{yy} * I_{\omega y}) / (I_{xy}^2 - I_{xx} * I_{yy}) \quad (\text{Eq. 1})$$

$$Y_{ref} = (I_{xx} * I_{\omega x} - I_{xy} * I_{\omega y}) / (I_{xy}^2 - I_{xx} * I_{yy}) \quad (\text{Eq. 2})$$

If we apply them to mono- symmetric sections by setting  $I_{xy} = 0$ , then they can be simplified into

**Case 3:**

$$X_{ref} = I_{\omega y} / I_{xx} \quad (\text{Eq. 3})$$

$$Y_{ref} = - I_{\omega x} / I_{yy} \quad (\text{Eq. 4})$$

**It happens** that in some textbooks and/or references where you may find Eq. 3 and Eq. 4 given as the formulas for shear center calculation with **no mentioning** that they are only applicable to mono-symmetric profiles.

If we were “new to the task” of locating shear center without further checking and blindly following (applying) Eq. 3 and Eq. 4 to a section having no axes of symmetry, we would still be able to obtain  $X_{ref}$  and  $Y_{ref}$  but in fact these formulas (results) were **wrong** for the occasion. We must be **careful** about the **trap** I mentioned in the first paragraph on this matter. If you plan to **automate** your calculation I recommend that you always use Eq. 1 and Eq. 2.

Finally by parallel shifting w.r.t. centroid, the coordinates of shear center per user-axes are:

$$x_o = X_{ref} + c_x$$

$$y_o = Y_{ref} + c_y$$

### Double Sectorial Area w.r.t. Shear Center ( $\omega_o$ )

Our next assignment per Calc step 14 (Calc page 11) is to perform integration for  $\omega_o$ , which is the “DSA w.r.t. Shear Center”. In its integration formula  $\omega_o = \int \rho_o ds$ , the term  $\rho_o$  looks familiar but is actually a new term.

Let’s go back to the formula for DSA w.r.t. centroid  $\omega = \int \rho ds$  and recall that  $\rho$  is the normal distance from **centroid** to the segment tangent. Here the only difference is that  $\rho_o$  is used in place of  $\rho$ , where  $\rho_o$  is the normal distance from **shear center** to the segment tangent. Obviously the formulas for figuring out the geometry of  $\rho$  and  $\rho_o$  resemble each other if you check the Calc.

This is the second process involving shear flow-related data. Starting with the same shear flow branch definitions, the numerical procedure for  $\omega_o = \int p_o ds$  is “almost identical” to that as described in the calculation for  $\omega = \int p ds$ . A worked-out “Node-driven” data flow for shear flow branch #1 can be traced by following the dotted arrows as shown on Calc page 12. I would leave it up to you to “see” any difference between  $\omega$  and  $\omega_o$ .

<b>SD</b> Structural Design Corporation		Calcs. For <u>WARPING CONSTANT OF GIRDER</u> <u>W21x55 + C8x11.5</u>	Calc. No.
Client		Prepared by <u>John Feng</u>	Rev.
Project <u>SDC STANDARD</u>		Reviewed by	Date
			Page 14 of 16

16. PROFILE WARPING CONSTANT ( $C_w$ )

$$C_w = \int \omega_n^2 ds = \frac{1}{3} \sum (\omega_{ni}^2 + \omega_{ni}\omega_{nj} + \omega_{nj}^2)(t ds)$$

SEGMENT ID	N O D E		$\omega_n$		$\omega_{ni}^2 + \omega_{ni}\omega_{nj} + \omega_{nj}^2$	t ds (A)	$\frac{A}{3}(\omega_{ni}^2 + \omega_{ni}\omega_{nj} + \omega_{nj}^2)$
	FROM i	TO j	$\omega_{ni}$	$\omega_{nj}$			
10	10	11	-2.6451	-2.9692	23.666	0.677	5.3408
11	11	12	-2.9692	-3.4325	30.7895	0.9677	9.9319
12	12	13	-3.4325	-3.9868	41.3613	0.0002	0.0027
13	13	14	-3.9868	-4.3636	52.332	0.825	14.3913
14	14	15	-4.3636	-18.2313	430.9373	1.43	205.4308
15	15	16	-18.2313	-29.5434	1450.5323	0.754	364.5671
21	6	21	-0.0523	-41.9545	1762.3797	1.43	840.0676
22	21	22	-41.9545	-64.0485	8549.5125	0.754	2148.7771
31	12	31	-3.4325	7.3844	40.9649	0.8371	11.4306
32	31	32	7.3844	18.7344	543.8488	0.8783	159.2299
33	32	33	18.7344	36.2864	2347.4815	0.8385	656.1209
4	4	5	63.9439	41.85	8516.2946	0.754	2140.4283
41	14	41	-4.3636	9.5042	67.8981	1.43	32.3648
42	41	42	9.5042	16.8163	532.9412	0.754	133.9459
5	5	6	41.85	-0.0523	1749.2359	1.43	832.8024
6	6	7	-0.0523	-0.8982	0.8565	1.8525	0.5289
7	7	8	-0.8982	-2.1996	7.621	2.85	7.2399
8	8	9	-2.1996	-2.7802	18.683	1.2713	7.9169
9	9	10	-2.7802	-2.6451	22.0792	0.0002	0.0014
PROFILE WARPING CONSTANT $C_w = 7571.5193$ ← SUM							

## Normalized Unit Warping ( $\omega_n$ ) and Warping Constant ( $C_w$ )

“Normalized Unit Warping” is designated as  $\omega_n$ . The formula for  $\omega_n$  expressed in algebraic terms is shown on Calc page 13. It entails summations of nodal property “ $\omega_o$ ” and the cross sectional area of segment “t ds”. This would require a two-step procedure to arrive at  $\omega_o$  because one of the two (2) terms is a profile “constant” and the other is a nodal “variable”. The first step involves a “Segment-driven” process that integrates ( $\omega_o$  t ds) of each segment over the entire profile into a constant “C”. The second step is simply a “Node-driven” process that subtracts “ $\omega_o$ ”

from “C” to arrive at  $\omega_n$  at each node as shown on Calc page 13. The procedure for calculating warping constant becomes fairly simple once the corresponding values of “t ds” and “ $\omega_n$ ” have been:

- “collected” from all nodes and
- “grouped” by incidence nodes with respective to each segment.

Using a SELECT Query for the “grouping” followed by a “Summation”, we finally complete the calculation of  $C_w$  constant for our example by way of “classic method” (see resulting tabulations per Step 16 on Calc page 14). We call it a “classic method” because the undertaking was achieved by performing numerical integration for “formulas” derived from “classic torsion theory”. Here are some of the disadvantages from using “classic solution method” as demonstrated:

- (a) tedious and time consuming
- (b) confusing every so often
- (c) full of numerical traps throughout the process.


Besides classic solution method there are other methods we can use to avoid all that. Some of these are so called the “direct method” or “displacement method”. Without knowing anything about principal axes, the “direct” formulation would lead to solution of “ $x_o$ ”, “ $y_o$ ” and all the nodal  $\omega_n$  values in one shot. Better yet we don’t even need “shear flow definition” for that. This sounds fantastic and is a good trade off for not having to mess with numerical integration. However the most cumbersome chores in implementing these methods are: (1) setting up the property-geometry matrices and (2) solving for the ensuing simultaneous equations.

Applying direct method to solve for  $\omega_n$  and shear center coordinates for our example profile of 20 nodes would require inversion of a 22 X 22 matrix.

### Warping Static Moment ( $S_w$ )

“Warping Static Moment” (**WSM**) is another warping-related property designated as  $S_w$ . Values of **WSM** must be known at each node before the “state of warping shear stress” of a profile can be evaluated. Even if your interest is in  $C_w$  or  $\omega_n$  only and you have no desire to go any further, we still recommend that you follow through with this important subject if your member has arbitrary profile geometry.

In the simplest case for sections having doubly- or mono- symmetric profile, we can easily verify if the nodal  $\omega_n$  values are distributed consistently and evenly about the matching axis of symmetry to



Calcs. For WARPING CONSTANT OF GIRDER  
W21x55 + C8x11.5

Calc. No.	
Rev	Date
Page <u>15</u>	of <u>16</u>

Client \_\_\_\_\_

Project SDC STANDARD

Prepared by JOHN FONG

Date 7-6-10

Reviewed by \_\_\_\_\_

Date \_\_\_\_\_

# 17. INTEGRATION FOR WARPING STATIC MOMENT ( $S_w$ )

$$S_w = \int_0^s \omega_n t ds$$

FOR ANY TYPICAL SEGMENT WITH SHEAR FLOWS FROM NODE  $i$  TO NODE  $j$

$$S_{wi} = \sum S_w \text{ @ NODE } i \text{ PRIOR TO INTEGRATION}$$

$$S_{wj} = S_{wi} + \frac{\omega_{ni} + \omega_{nj}}{2} \cdot t ds$$

$$= (\omega_{ni} + \omega_{nj}) (t ds / 2) + S_{wi}$$

K

PROCEED WITH  $S_w$  INTEGRATION IN ORDER EXACTLY REVERSE TO THE NODE AND BRANCH SEQUENCE USED FOR INTEGRATION OF SECTORIAL AREAS:

BRANCH ID	NODE			SEGMENT			K	$S_{wi}$	$S_{wj}$
	SEQUENCE	ID	INCIDENCE	$\omega_{ni,j}$	$\omega_{ni} + \omega_{nj}$	$t ds / 2$			
4	3	4	4	63.9439			BRANCH INITIALIZING $\rightarrow 0$		
	2	5	4	41.85	$\rightarrow 105.7939 \times 0.377 = 39.8843$			$0$	$39.8843$
	1	6	5	-0.0523	$\rightarrow 41.7977 \times 0.715 = 29.8854$			$39.8843$	$69.7696$
3	4	33	33	36.2864			BRANCH INITIALIZING $\rightarrow 0$		
	3	32	33	18.7344	$\rightarrow 55.0208 \times 0.4192 = 23.0674$			$0$	$23.0674$
	2	31	32	7.3844	$\rightarrow 26.1188 \times 0.4392 = 11.4714$			$23.0674$	$34.5382$
	1	12	31	-3.4325	$\rightarrow 3.9519 \times 0.4186 = 1.6543$			$34.5382$	$36.1925$
2	11	22	22	-64.0485			BRANCH INITIALIZING $\rightarrow 0$		
	10	21	22	-41.9545	$\rightarrow 106.003 \times 0.377 = -39.9631$			$0$	$-39.9631$
	9	6	21	-0.0523	$\rightarrow -42.0068 \times 0.715 = -30.0349$			$-39.9631$	$-69.998$
	8	7	6	-0.8982	$\rightarrow -0.9505 \times 0.9262 = -0.8804$			$-69.998$	$-70.8782$
	7	8	7	-2.1996	$\rightarrow -3.0978 \times 1.425 = -4.4144$			$-0.8804$	$-71.7626$
	6	9	8	-2.7802	$\rightarrow -4.9798 \times 0.6356 = -3.1652$			$-4.4144$	$-75.9278$

Associated Printers, Inc. • (708) 367-1009



“see” immediately if we “made any mistake”. But for sections with no axis of symmetry, there is no easy way (or no way) to validate the adequacy of warping-related properties (both  $\omega_n$  and  $C_w$ ). Warping constant  $C_w$  comes from a chain reaction involving a full course of profile properties, to name a few:

centroid, neutral axes, user-axes moment of inertia, principal-axes moment of inertia, shear flow data, shear center,  $\rho$ ,  $\rho_o$ ,  $\omega$ ,  $\omega_o$  and  $\omega_n$ , etc.

$\omega_n$  are the very last set of section properties we “touched” prior to final summation of  $C_w$ . It is impossible to tell how right (or wrong) is our “ $C_w$ ” by looking at each  $\omega_n$ , but we can always compare them with values of **WSM** to “see” if we “made any mistake”.

From textbooks on this subject we ran into a generic formula for **WSM** that looks something like this:

$$S_w = \int \omega_n t ds$$

which is a definite integral ranging from “0” to “s” (where “s” is the coordinate of some node along the middle line of profile).

**WSM** seems to only exist as “Cold Hard Figures” in graphical plots appearing in textbooks. Scarcely there is any “practical” procedure accompanied with worked-out sample problem(s). “Textbooks” have too much to cover therefore we can’t blame their authors for concentrating too much on the “why” and not so much about the “how” for any particular subject(s). After all, most examples are suitable for students in pursuing passing grades. Offering “Cold Hard Figures” without practical “step-by-step procedures” don’t go very far in helping (already graduated) engineers trying to solve everyday problems.

How to implement the integration “ $S_w = \int \omega_n t ds$ ” becomes a mystery to a lot of engineers. We did it our way here and would like to share that with you. Hopefully it fills in the void before you find some other (more elegant) solution algorithms buried somewhere (we reckon).

Let us break into the subject by apprehending a few important facts:

- (a) “warping normal stresses” are proportional to  $\omega_n$
- (b) “warping shear stresses” are proportional to  $S_w$
- (c) “warping shear stresses” vanish where “warping normal stresses” peak
- (d) “warping normal stresses” always peak at terminal ends (nodes)
- (e) “warping shear stresses” equal to zero at terminal nodes and peak where “warping normal stresses” vanish
- (f) therefore  $S_w$  equals to zero at terminal nodes

These can be verified by inspecting the “ $\omega_n$ ” (already known) and “ $S_w$ ” (not yet calculated) values as plotted on Calc page 16. We only need to compare the respective magnitude at terminal nodes 16, 42, 33, 4 and 22 to substantiate if we erred.

We mentioned earlier that the “shear flow information” would be used for at least three (3) times:

- (a) first: “ $\omega = \int \rho ds$ ” integral of **DSA** w.r.t. centroid,
- (b) second: similarly the “ $\omega_o = \int \rho_o ds$ ” integral of **DSA** w.r.t. shear center,
- (c) and now the third: somewhat differently the “ $S_w = \int \omega_n t ds$ ” integral of **WSM**.

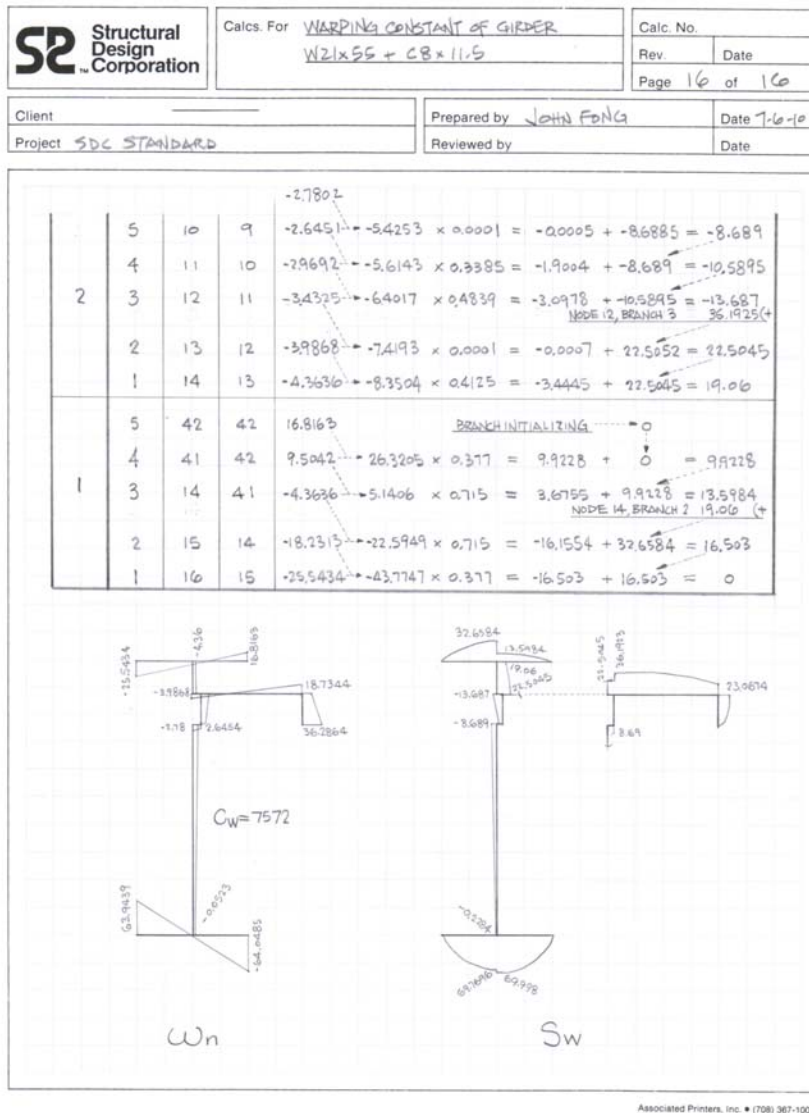
The word “similarly” and the phrase “somewhat differently” refer to “how we sequence” the “shear flow branch information” during integration. The path taken on for DSA integration follows an “ascending flow” while in WSM it follows a “descending flow”. Ascending flow starts from the first branch (#1), first node (#1) and ends at the last branch (#4), last node (#3). Descending flow proceeds along a path in exactly the reverse order.

It's easy to implement definite integral ranges from “0” to “s” for profiles conform to singular continuity (channels for example). Each of these profiles has only two (2) ends for us to concern about. We simply carry out numerical integration either from one end to the other or vice versa. It

is convenient to interpret “0” as one of the terminal ends and “s” being the opposite end (or somewhere in the middle). It doesn't matter whether we follow ascending or descending order because there is no wrong end.

Problem is how to deal with profiles with multiple terminal ends. If we have not yet gone through Calc pages 15 and 16, we may ask these questions:

- how to implement the definite integral ranges from “0” to “s” for the example profile with five (5) terminal ends?
- which node to start with?
- where does the integration end?



Answers to all these questions are: “Use descending flow”. Here is where it deserves more thorough explanation. “Descending flow” is “somewhat different” from “ascending flow” in the way we handle the “s” portion of the integral. As an excise we would leave it up to you to “see” the difference when this is all said and done.

We recognize that “s” is the intermediate “integration path taken” up to somewhere within the profile boundary. The leading extremity of the “s path” may arrive at anywhere in the vicinity of any given Node. Chances are that it may be (a) right on, (b) nearby or (c) far away from some terminal end in focus. Making things simple we purposely let “s” be “right on” a certain segment's

“From Node” (or “To Node”). From a numerical integration standpoint, it’s easier to deal with start node index “i” and end node index “j” than using the adjectives “From” and “To” (although either “i” or “j” can coincide with a “From” or “To” node in a physical sense). In terms of integration we would always handle the “segment **WSM**” sequence from “node i” to “node j”. The path we take must follow the “**Descending Flow**” along any given branch. The calculated values of **WSMs** for a typical segment are labeled as  $S_{wi}$  for node “i” and  $S_{wj}$  at node “j”, respectively.

Let’s go through the  $S_w$  algorithm:

- (a) process the integration steps (b) through (d) according to the descending flow branch sequence (ex: 4, 3, 2 then 1). In other words, always finish all the “twigs” first and the “trunk” last,
- (b) when starting a new branch (from a terminal node), always set  $S_{wi} = 0$ ,
- (c) when starting a typical segment,
  - a. verify “proper flow **vector**” based on whether the “i-j flow” for integration is oriented along the “From-To flow” or along the “To-From flow”,
  - b. identify which is node “i” and which is node “j”
  - c. look up the value of  $\omega_n$  at node “i” as  $\omega_{ni}$
  - d. set  $S_{wi}$  as the **summation** of all  $S_{wi}$  already known at node “i”,
- (d) to finish up a typical segment at node “j”,
  - a. look up the value of  $\omega_n$  at node “j” as  $\omega_{nj}$
  - b. look up the value of cross sectional area of segment as “tds”
  - c. calculate  $K = (\omega_{ni} + \omega_{nj}) * (tds) / 2$
  - d. set  $S_{wj} = K + S_{wi}$

The worked-out “**Node-driven**” data flow following  $S_w$  algorithm for the entire profile can be traced by the dotted arrows appearing in the tabulation shown on Calc pages 15 and 16. Because  $S_w$  must equal to zero at terminal nodes, it is easy to verify if we “**made any mistake**”. Most importantly we need to make sure  $S_{wj} = 0$  (or approaches zero) at the very last node we processed (i.e. shear flow node serial #1, flow branch #1, or input node 16 on segment 15). If “ $S_w < 0$ ” or “ $S_w$  does not approach zero”, there is definitely “something wrong” during the process. This has demonstrated that validation of  $C_w$  for “profiles with no axis of symmetry” would not be possible without helps from  $S_w$ .

It is worthwhile knowing the distribution of WSM throughout the profile is non-linear with intermittent peaks and valleys (**PAVS**). To pin point where the PAVS are by inspecting  $S_{wi}$  and  $S_{wj}$  at both ends of segments alone is not enough. Because WSM peaks where “warping normal stress” vanishes, it is easy to locate the PAVS by checking on each segment whether if the product of  $\omega_{ni} * \omega_{nj} < 0$ . If this condition is true then we can locate the point of PAVS by interpolation and calculate the PAVS accordingly. This extra step for PAVS is not necessary unless our interest also include the state of warping shear stress.

## Conclusion

The subject in dealing with flexural horizontal shear stress (**FHSS**) was absent in this article until now for it has nothing to do with warping constant. FHSS is calculated using “VQ/It” formula found in many textbooks on “strength of material”. Property “Q” is called “First moment” or “Moment of

area". When a profile is subjected to both X- and Y- forces, the FHSS should be calculated using both  $Q_x$  and  $Q_y$ . To figure out  $Q_x$  and  $Q_y$  for sections of arbitrary profile geometry with no axis of symmetry it requires numerical integration w.r.t. the principal axes in steps similar to but slightly different from algorithm used for " $S_w$ ". It should be not so hard to figure out the difference.

In summary, to calculate the **warping constant** for this example profile with 20 nodes, we probably have to deal with at least 2000 numerical entries plus mathematical operators, parentheses and sign change keys, etc. Through it all you may have sensed that the demand in data management rivals the effort in number crunching. Fine-tuning numerical accuracy against excessive rounding-off and truncation errors is a tricky chore in itself. An engineering problem suddenly eases itself into a data management problem without realizing it. Therefore the solution process for warping constant is a perfect candidate for automation using database approach.