An example of looks like this

contour, image, levels=[30.,40.], path_info=pinfo, path_xy=pxy, /path_data_coords, /closed

Here, pinfo is a structure that tells you about the information contained in pxy.
You have to write some flexible code to handle this properly because pxy is a single, 2xN array
that actually contains all of the contours you requested, strung together vertically. pxy[0,*] are the
x values, and pxy[1,*] are the y values.

To make it usable, you have to sort through pinfo. pinfo is an array of structures with fields {type,
high_low, level, n, offset, value}. The pinfo.offset field elements tell you the index values of pxy
where each contour starts, and pinfo.n elements tell you how many points are in each contour.

Here is one array for each contour.
xy0 = pxy[0:1, pinfo[0].offset : pinfo[0].offset + pinfo[0].N - 1]
xy1 = pxy[0:1, pinfo[1].offset : pinfo[1].offset + pinfo[1].N - 1]

Where you have to be careful, I've found, is that sometimes you get two (or more?) contours for
the same level value. In my example I have 2 levels, but you could end up with 3 or more contour
"pieces." So check the pinfo.value field to see if you have separate contours with matching level
values, and group/concatenate them if necessary.

-Ken