Dear Steve,

Below I have tried to describe how the routine works. If it is still difficult to understand I could send some figures that should make it clearer with Fax.

Regards,

Peter

We would like to count the number of sides in the Voroni (is that the correct name?) polygons. With paper and pen that is easily done: Focus on one particle (vortex) and make straight lines to a few fairly near neighbors. Bisect each such line by a perpendicular line. Some of these perpendicular lines will now make up a polygon. Count the number of sides in this polygon.

To make a simple algorithm out of this we focus on one particle at the time and choose the origin to be the position of that particle. We then let $\mathbf{r}_i$ denote the positions of a number of (≈ 10) particles that are close to the origin. The question is how many particles that contribute to the polygon. Note that the pen and paper method discussed above focuses on $\mathbf{r}_i/2$ instead of $\mathbf{r}_i$ because of our bisecting the line connecting the origin and $\mathbf{r}_i$. This is, however, not necessary. We may just as well draw lines through $\mathbf{r}_i$ that are perpendicular to $\mathbf{r}_i$. Below we first discuss a method that over-simplifies the problem, and then turn the real algorithm.

For each particle j we examine if its associated line contributes to the polygon. To do that we loop over particles i and examine whether $\mathbf{r}_j$ is beyond the line associated with $\mathbf{r}_i$ or not. If it is beyond such a line it cannot contribute to the polygon. This information is given by the quantity

$$S_{ij} = \mathbf{r}_i \cdot (\mathbf{r}_i - \mathbf{r}_j).$$

If $\mathbf{r}_j$ is beyond the line perpendicular to $\mathbf{r}_i$ then $S_{ij} > 0$. The condition for $\mathbf{r}_j$ to contribute to the polygon is therefore $S_{ij} < 0$ for all $i \neq j$.

The above condition is, however, too restrictive. For $\mathbf{r}_j$ to contribute to the polygon it is instead enough that the above is true for a part of the line associated with $\mathbf{r}_j$. Introduce q to parametrize the line associated with $\mathbf{r}_j$,

$$\mathbf{r}_j(q) = \mathbf{r}_j + q\mathbf{r}_j^{\perp}.$$

Here $\mathbf{r}_j^{\perp}$ is a vector that is perpendicular to $\mathbf{r}_j$. With the notation $\mathbf{r}_j = (x_j, y_j)$ we can write $\mathbf{r}_j^{\perp} = (y_j, -x_j)$. The question is now if there is some interval in

q for which $S_{ij}(q) < 0$ (with an obvious generalisation of the above equation) for all i.

To answer that question we note that each point $\mathbf{r}_i$ with $\mathbf{r}_j^\perp \cdot \mathbf{r}_i \neq 0$ gives either an upper or a lower bound on q (depending on the sign of $\mathbf{r}_j^\perp \cdot \mathbf{r}_i$). In both cases, the limiting value for q is the one that makes $S_{ij}(q) = 0$:

$$0 = \mathbf{r}_i \cdot (\mathbf{r}_i - \mathbf{r}_j - q\mathbf{r}_j^\perp) = S_{ij} - q\mathbf{r}_i \cdot \mathbf{r}_j^\perp$$

that gives

$$q = S_{ij}/(\mathbf{r}_i \cdot \mathbf{r}_j^\perp)$$

In the program the lower and upper bounds of q are determined by a loop over i. If it turns out that $q_{\mathrm{lo}} \geq q_{\mathrm{hi}}$ it is concluded that there is no part of the line associated with $\mathbf{r}_j$ that is within (= closer to the origin than) all the lines associated with the other particles. Therefore the line associated with $\mathbf{r}_j$ doesn't belong to the polygon. (The above condition could instead be chosen to $q_{\mathrm{lo}} > q_{\mathrm{hi}}$. The case when four lines meet at the same point is then treated differently.)

The above description should be easy to compare with the C-code with the identification `sum`$=S$ and `vvperp`$=\mathbf{r}_i \cdot \mathbf{r}_j^\perp$.