I'll tell a few words about curve-curve extrema here.
Two curves are given: $C_1(U)$, $C_2(V)$. Extrema is a line which joins point on C_1 curve with point on curve C_2 and has extremum of length (locale maximum or locale minimum). If we stand $F(U, V)$ as a function of distance between points of given curves $C_1(U)$ and $C_2(V)$, then extrema joins the points $C_1(U_0)$ and $C_2(V_0)$, where U_0 and V_0 are root of system (U and V are unknown variables):

$$\begin{cases} \dfrac{\partial F}{\partial U} = 0 \\ \dfrac{\partial F}{\partial V} = 0 \end{cases} \quad (1)$$

System (1) is solved by minimize of function $G(U,V) = \left(\dfrac{\partial F}{\partial U}\right)^2 + \left(\dfrac{\partial F}{\partial V}\right)^2 \rightarrow \min$. Minimize is made by gradient-method. Gradient determines a direction of minimum search. After that several steps are done in found direction (new gradients are not computing meanwhile). So, however in singularity point gradient is indeterminate, we can get into singularity point all the same.

Simple script illustrates all the said above:

```
Draw[]> bsplinecurve bc1 2 4 0.5 3 1 2 1.5 2 2 3 -3 2 -1 1 -1 2 -1 1 -1 3 -1 1 -1 3 -1 1 -1 3 -1 1 -1 2 -1 1 5 2 -1 1
Draw[]> bsplinecurve bc2 2 4 1 3 2 2 3 2 4 3 -3 6 2 1 1 6 2 1 1 5 2 1 1 5 2 1 1 5 2 1 1 6 2 1 5 6 2 1
Draw[]> extrema bc1 bc2
```

Some of found extremas are line which joins singularity points of curves **bc1** and **bc2**.

Following example shows a case, when line which joins singularity points is not extrema

```
Draw[]> bsplinecurve bc1 3 3 2 4 3 3 4 4 -3 2 4 1 0 11 -6 1 2 8 -4 1 2 8 -4 1 2 8 -4 1 0 11 -6 1 4 5 2 1
Draw[]> bsplinecurve bc2 3 3 2 4 3 3 4 4 -2 3 6 1 6 1 0 1 4 4 -2 1 4 4 -2 1 4 4 -2 1 6 1 0 1 5 7 -4 1
Draw[]> extrema bc1 bc2
```