

```
001 void sample(float dl, float ul, float pl, float cl,
002             float dr, float ur, float pr, float cr,
003             const float pm, const float um,
004             float &d, float &u, float &p)
005 {
006     float c, cml, cmr, pml, pmr, shl, shr, sl, sr, stl, str;
007
008     if (0.0 <= um)
009     {
010         if (pm <= pl)
011         {
012             shl = ul - cl;
013
014             if (0.0 <= shl)
015             {
016                 d = dl;
017                 u = ul;
018                 p = pl;
019             }
020             else
021             {
022                 cml = cl * pow(pm / pl, G1);
023                 stl = um - cml;
024
025                 if (0.0 > stl)
026                 {
027                     d = dl * pow(pm / pl, 1.0 / GAMA);
028                     u = um;
029                     p = pm;
030                 }
031                 else
032                 {
033                     u = G5 * (cl + G7 * ul);
034                     c = G5 * (cl + G7 * ul);
035                     d = dl * pow(c / cl, G4);
036                     p = pl * pow(c / cl, G3);
037                 }
038             }
039         }
040         else
041         {
042             pml = pm / pl;
043             sl = ul - cl * sqrt(G2 * pml + G1);
044
045             if (0.0 <= sl)
046             {
047                 d = dl;
048                 u = ul;
049                 p = pl;
050             }
051             else
052             {
053                 d = dl * (pml + G6) / (pml * G6 + 1.0);
054                 u = um;
055                 p = pm;
056             }
057         }
058     }
059     else
060     {
061         if (pm > pr)
062         {
063             pmr = pm / pr;
064             sr = ur + cr * sqrt(G2 * pmr + G1);
065
066             if (0.0 >= sr)
067             {
068                 d = dr;
069                 u = ur;
070                 p = pr;
071             }
072             else
073             {
074                 d = dr * (pmr + G6) / (pmr * G6 + 1.0);
075                 u = um;
076                 p = pm;
077             }
078         }
079         else
080         {
081             shr = ur + cr;
082
083             if (0.0 >= shr)
084             {
085                 d = dr;
086                 u = ur;
087                 p = pr;
088             }
089             else
090             {
091                 cmr = cr * pow(pm / pr, G1);
092                 str = um + cmr;
093
094                 if (0.0 <= str)
095                 {
096                     d = dr * pow(pm / pr, 1.0 / GAMA);
097                     u = um;
098                     p = pm;
099                 }
100                 else
101                 {
102                     u = G5 * (-cr + G7 * ur);
103                     c = G5 * (cr - G7 * ur);
104                     d = dr * pow(c / cr, G4);
105                     p = pr * pow(c / cr, G3);
106                 }
107             }
108         }
109     }
110 }
```