

```

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

```