```
diff --git a/src/lpj/photosynthesis.c b/src/lpj/photosynthesis.c
index 1cc35247..061f5293 100644
--- a/src/lpj/photosynthesis.c
+++ b/src/lpj/photosynthesis.c
@@ -43,7 +43,7 @@
  *
  */

-Real photosynthesis(Real *agd,      /**<  total daytime net photosynthesis (gC/m2/day */
+Real photosynthesis(Real agd[2],    /**<  total daytime net photosynthesis (gC/m2/day */
                     Real *rd,       /**< daily leaf respiration (gC/m2/day) */
                     int path,       /**< Path (C3/C4) */
                     Real lambda,
@@ -56,10 +56,11 @@ Real photosynthesis(Real *agd,      /**<  total daytime net photosynthesis (gC/m2
 {
   static Real ko,kc,tau,pi,c1,c2;
   Real je,jc,phipi,and,adt,b,s,sigma,vm;
+  Real c1d,c2d,jed,jcd;
   static Real fac,gammastar;
   if(tstress<1e-2)
   {
-    *agd=0;
+    agd[0]=agd[1]=0;
     *rd=0;
     return 0;
   }
@@ -96,13 +97,15 @@ Real photosynthesis(Real *agd,      /**<  total daytime net photosynthesis (gC/m2
       /*      Recalculation of C1C3, C2C3 with actual pi */

      c1=tstress*param.alphac3*((pi-gammastar)/(pi+2.0*gammastar));
-
+     c1d=tstress*param.alphac3*co2*(1./(pi+2*gammastar)-(pi-
gammastar)/(pi+2*gammastar)/(pi+2*gammastar));
      c2=(pi-gammastar)/(pi+fac);
+     c2d=co2/(pi+fac)-co2*(pi-gammastar)/(pi+fac)/(pi+fac);
     }
     else /* C4 photosynthesis */
     {
       c1=tstress*param.alphac4;
       c2=1.0;
+      c2d=0;
       b=param.bc4;
       s=(24/daylength)*param.bc4;
       sigma=1-(c2-s)/(c2-param.theta*s);
@@ -118,7 +121,12 @@ Real photosynthesis(Real *agd,      /**<  total daytime net photosynthesis (gC/m2
        */
       phipi=lambda/lambdamc4;
       if(phipi<1)
+      {
        c1=tstress*phipi*param.alphac4;
+        c1d=tstress*param.alphac4/lambdamc4;
+      }
+      else
+        c1d=0;
     }

     /*
@@ -129,19 +137,26 @@ Real photosynthesis(Real *agd,      /**<  total daytime net photosynthesis (gC/m2
      *     Eqn 3, Haxeltine & Prentice 1996
      */
     je=c1*apar*cmass*cq/daylength;
+    jed=c1d*apar*cmass*cq/daylength;

     /*
      * Calculation of rubisco-activity-limited photosynthesis rate JC, molC/m2/h
      *     Eqn 5, Haxeltine & Prentice 1996
      */
     jc=c2*hour2day(vm);
+    jcd=c2d*hour2day(vm);

     /*
      *     Calculation of daily gross photosynthesis, Agd, gC/m2/day
      *     Eqn 2, Haxeltine & Prentice 1996
      */
```

```
-    *agd=(je+jc-sqrt((je+jc)*(je+jc)-4.0*param.theta*je*jc))/(2.0*param.theta)*daylength;
+    agd[0]=(je+jc-sqrt((je+jc)*(je+jc)-4.0*param.theta*je*jc))/(2.0*param.theta)*daylength;
+    if(je==0 && jc==0)
+      agd[1]=(jed+jcd)*daylength/2/param.theta;
+    else
+      agd[1]=(jed+jcd-((je+jc)*(jed+jcd)-2.0*param.theta*(jed*jc+je*jcd))/sqrt((je+jc)*(je+jc)-
4.0*param.theta*je*jc))*daylength/2.0/param.theta;
+    agd[1]*=8.314*degCtoK(temp)/p*1000.0/cmass;

     /*    Daily leaf respiration, Rd, gC/m2/day
      *    Eqn 10, Haxeltine & Prentice 1996
@@ -150,7 +165,7 @@ Real photosynthesis(Real *agd,     /**<  total daytime net photosynthesis
(gC/m2

     /*    Daily net photosynthesis (at leaf level), And, gC/m2/day */

-    and=*agd-*rd;
+    and=agd[0]-*rd;

     /*     Total daytime net photosynthesis, Adt, gC/m2/day
      *     Eqn 19, Haxeltine & Prentice 1996
diff --git a/src/lpj/water_stressed.c b/src/lpj/water_stressed.c
index 8245b52c..cad02490 100644
--- a/src/lpj/water_stressed.c
+++ b/src/lpj/water_stressed.c
@@ -14,7 +14,7 @@

 #include "lpj.h"

-#define EPSILON 0.1  /* min precision of solution in bisection method */
+#define EPSILON 0.001  /* min precision of solution in bisection method */

 typedef struct
 {
@@ -24,7 +24,7 @@ typedef struct

 static Real fcn(Real lambda,Data *data)
 {
-  Real agd,rd;
+  Real agd[2],rd;

   /*
    *             Call photosynthesis to determine alternative total
@@ -33,7 +33,7 @@ static Real fcn(Real lambda,Data *data)
    *             guess for lambda (xmid)
    */

-  return data->fac*(1-lambda)-photosynthesis(&agd,&rd,data->path,lambda,
+  return data->fac*(1-lambda)-photosynthesis(agd,&rd,data->path,lambda,
                                              data->tstress,data->co2,
                                              data->temp,data->apar,
                                              data->daylength);
@@ -47,6 +47,31 @@ static Real fcn(Real lambda,Data *data)

 } /* of 'fcn' */

+static void fcn_newton(Real res[2],Real lambda,Data *data)
+{
+  Real agd[2],rd;
+
+  /*
+   *             Call photosynthesis to determine alternative total
+   *             daytime photosynthesis estimate (adt2) implied by
+   *             Eqns 2 & 19, Haxeltine & Prentice 1996, and current
+   *             guess for lambda (xmid)
+   */
+  res[0] = data->fac*(1-lambda)-photosynthesis(agd,&rd,data->path,lambda,
+                                               data->tstress,data->co2,
+                                               data->temp,data->apar,
+                                               data->daylength);
+  res[1] = -data->fac-agd[1];
+  /*
+   *             Calculate total daytime photosynthesis implied by
+   *             canopy conductance from water balance routine and
+   *             current guess for lambda (xmid).  Units are mm/m2/day
+   *             (mm come from gpd value, mm/day)
+   *             Eqn 18, Haxeltine & Prentice 1996
```

```
+   */
+
+} /* of 'fcn' */
+
 Real water_stressed(Pft *pft, /**< pointer to PFT variabels */
                     Real aet_layer[LASTLAYER],
                     Real gp_stand,
@@ -64,16 +89,16 @@ Real water_stressed(Pft *pft, /**< pointer to PFT variabels */
                     Bool permafrost
                     ) /** \return gross primary productivity (gC/m2) */
 {
-  int l,i;
-  Real supply,supply_pft,demand,demand_pft,wr,lambda,gpd,agd,gc,aet,aet_cor;
+  int l,i,iter;
+  Real supply,supply_pft,demand,demand_pft,wr,lambda,gpd,agd[2],gc,aet,aet_cor;
   Data data;
   Real roots;
   Real rootdist_n[LASTLAYER];
   Real layer,root_u,root_nu;
   Real freeze_depth,thaw_depth;
   Bool isless=FALSE;
-
-  wr=gpd=agd=*rd=layer=root_u=root_nu=aet_cor=0.0;
+  Bool err;
+  wr=gpd=agd[0]=agd[1]=*rd=layer=root_u=root_nu=aet_cor=0.0;
   forrootsoillayer(l)
   rootdist_n[l]=pft->par->rootdist[l];
   if(permafrost)
@@ -201,14 +226,21 @@ Real water_stressed(Pft *pft, /**< pointer to PFT variabels */
     else
       data.apar = par*alphaa(pft)*pft->fapar;                      /** fapar
calculation of trees and grass already include albedo*/
     data.daylength=daylength;
-    lambda=bisect((Bisectfcn)fcn,0.02,LAMBDA_OPT+0.05,&data,0,EPSILON,10);
-    photosynthesis(&agd,rd,data.path,lambda,data.tstress,data.co2,
+    if(fcn(0.02,&data)*fcn(LAMBDA_OPT+0.05,&data)<0)
+    {
+      lambda=newton((Newtonfcn)fcn_newton,0.5,&data,EPSILON,0,20,&iter,&err);
+      pft->stand->cell->output.count++;
+      pft->stand->cell->output.iter+=iter;
+    }
+    else
+      lambda=(0.02+LAMBDA_OPT+0.05)*0.5;
+    photosynthesis(agd,rd,data.path,lambda,data.tstress,data.co2,
                    temp,data.apar,daylength);
     /* in rare occasions, agd(=GPP) can be negative, but shouldn't */
-    agd=max(0,agd);
+    agd[0]=max(0,agd[0]);
     *rd=*rd;    /* DON'T DELETE THIS LINE */
   }
   else
     return 0.0;
-  return agd;
+  return agd[0];
 } /* of 'water_stressed' */
```