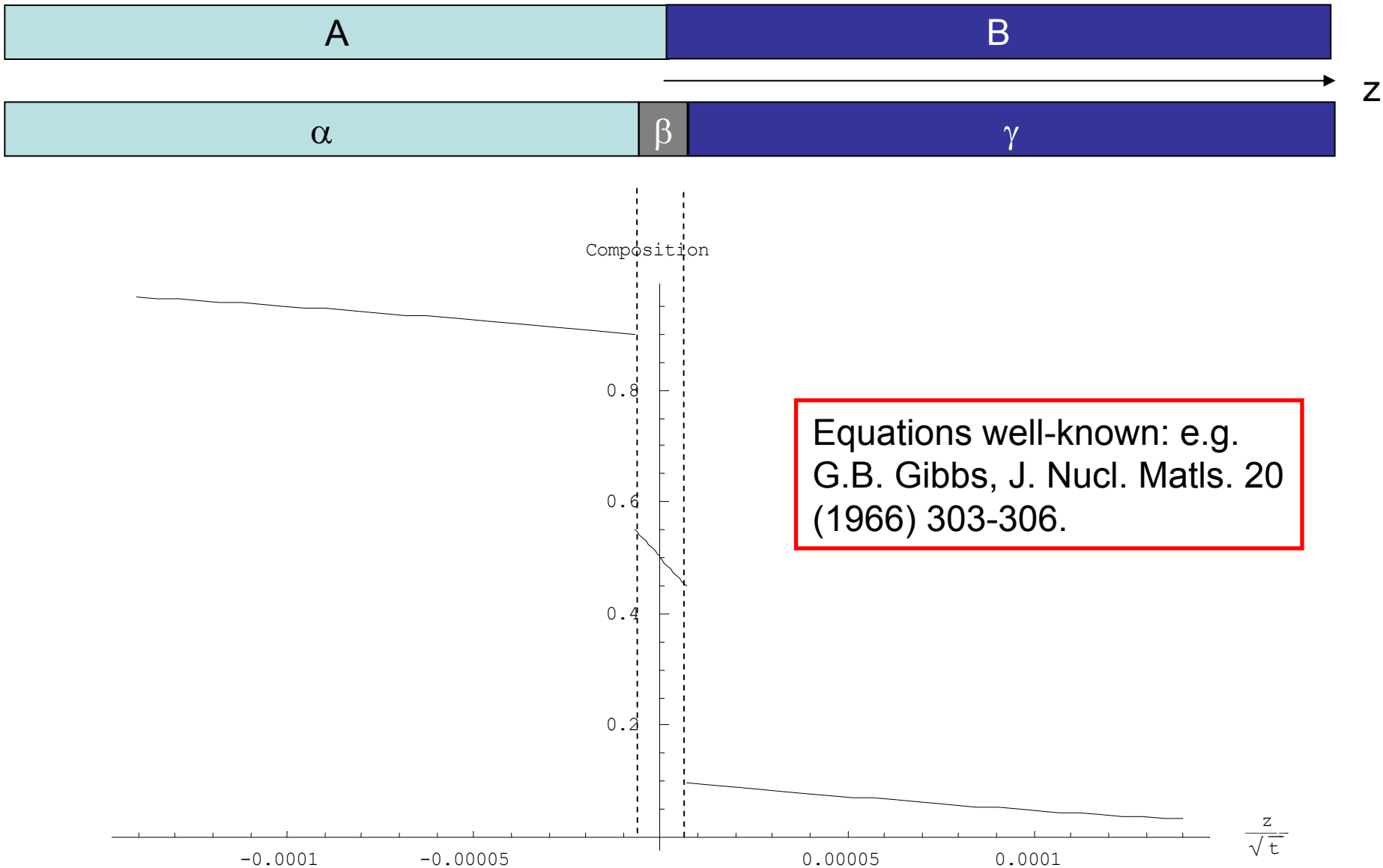
Mathematica Script for $t^{1/2}$ Growth of Planar Layers in Binary alloy

W.J. Boettinger, NIST



Equations well-known: e.g. G.B. Gibbs, J. Nucl. Matls. 20 (1966) 303-306.

Solution to Eqns. 7 of G.B. Gibbs, "Diffusion Layer Growth in a Binary System," J. Nucl.Matls. 20 (1966) 303 - 306.
by W. J. Boettinger, Metallurgy Division, NIST

The script treats the motion of two planar interfaces between three phases, alpha (1), beta (2) & gamma (3),
all growing from an initial interface at z =
0 between two initially uniform semi - infinite compositions of C01 and C30 respectively. Local equilibrium is assumed at the interphase
interfaces with compositions values given by : alpha / beta = C12 / C21,
beta / gamma C23 / C32. The two moving interface posions are z1 and
z2 .

Note : Beware ! This work done initially on February 19, 2004
seems is too dependent on the guess (or fails to converge) if the diffusion constants are more than 1 or 2 orders of magnitude apart. Need to fix!
I use FindRoot using two schemes . See if both give same values for K1 and K2. Probably need to expand Exp[x] Erf[x] for small x.

```
D1 = 1 * 10^-8;
D2 = 1 * 10^-9;
D3 = 1 * 10^-8;

C01 = 1;
C12 = 90 * 10^-2;
C21 = 55 * 10^-2;
C23 = 45 * 10^-2;
C32 = 10 * 10^-2;
C30 = 0;
```

Interface positions are normalized wrt D2 : z1 = 2 K1 (D2 * t) ^1 / 2, z2 = 2 K2 (D2 * t) ^1 / 2

```
S1 = (C01 - C12) / (C12 - C21);
S2 = (C21 - C23) / (C12 - C21);
S3 = (C21 - C23) / (C23 - C32);
S4 = (C32 - C30) / (C23 - C32);
d12 = (D1 / D2) ^ (1 / 2);
d32 = (D3 / D2) ^ (1 / 2);

ISPi = 1 / Sqrt[Pi];
```

```
zero1[K1_, K2_] := -K1 * (1 + Erf[K1 / d12]) * Erf[K1, K2] + d12 * S1 * ISPi * Exp[-K1^2 / d12] * Erf[K1, K2] + (-S2) * ISPi * Exp[-K1^2] * (1 + Erf[K1 / d12])

zero2[K1_, K2_] := -K2 * Erfc[K2 / d32] * Erf[K1, K2] + d32 * (-S4) * ISPi * Exp[-K2^2 / d32] * Erf[K1, K2] + S3 * ISPi * Exp[-K2^2] * Erfc[K2 / d32]

FindRoot[{zero1[K1, K2] == 0, zero2[K1, K2] == 0}, {{K1, 0}, {K2, 0}}]
```

```
{K1 → -0.110791, K2 → 0.110791}
```

```
FindRoot[{zero1[K1, K2] == 0, zero2[K1, K2] == 0}, {{K1, -.2, -.1}, {K2, -.1, 0}}]
```

```
{K1 → -0.110791, K2 → 0.110791}
```

```
K1 = K1 /. %20;
K2 = K2 /. %20;
```

The Unscaled t^1/2 coefficients,
z1 = k1 * t^1/2; z2 = k2 * t^1/2 are:

```
k1 = 2 * Sqrt[D2] * K1
k2 = 2 * Sqrt[D2] * K2
```

$$-7.00703 \times 10^{-6}$$

$$7.00703 \times 10^{-6}$$

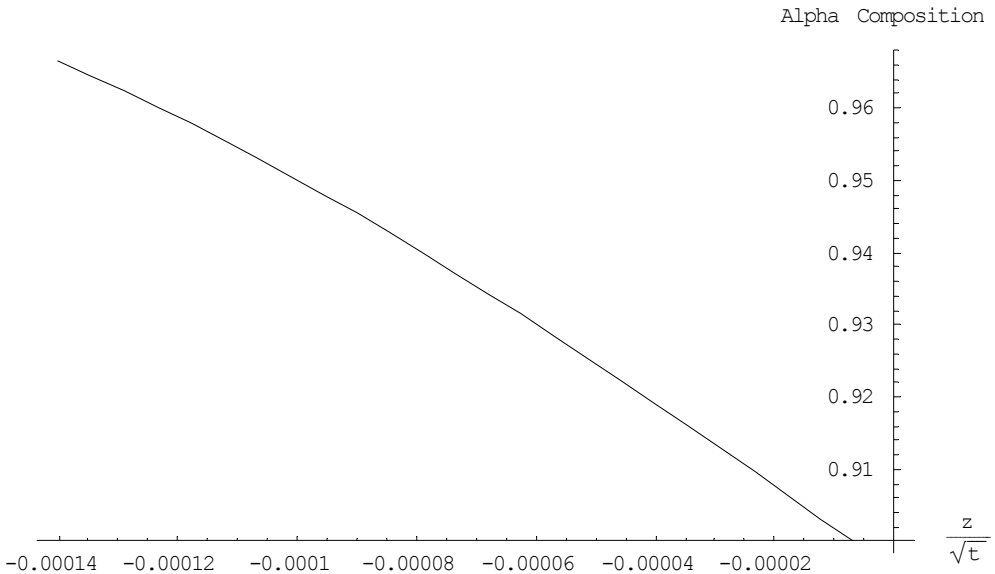
```
Calpha[eta_] := C01 - (C01 - C12) * (1 + Erf[eta / (4 * D1) ^0.5]) / (1 + Erf[K1 / d12])
Cbeta[eta_] := C21 - (C21 - C23) * (Erf[eta / (4 * D2) ^0.5] - Erf[K1]) / (Erf[K2] - Erf[K1])
Cgamma[eta_] := C30 + (C32 - C30) * (1 - Erf[eta / (4 * D3) ^0.5]) / (1 - Erf[K2 / 32])

leftside = 10;
rightside = 10;

kave = (k1 + k2) / 2;
delk = Abs[k1 - k2];
kmin = kave - leftside * delk;
kmax = kave + rightside * delk;
```

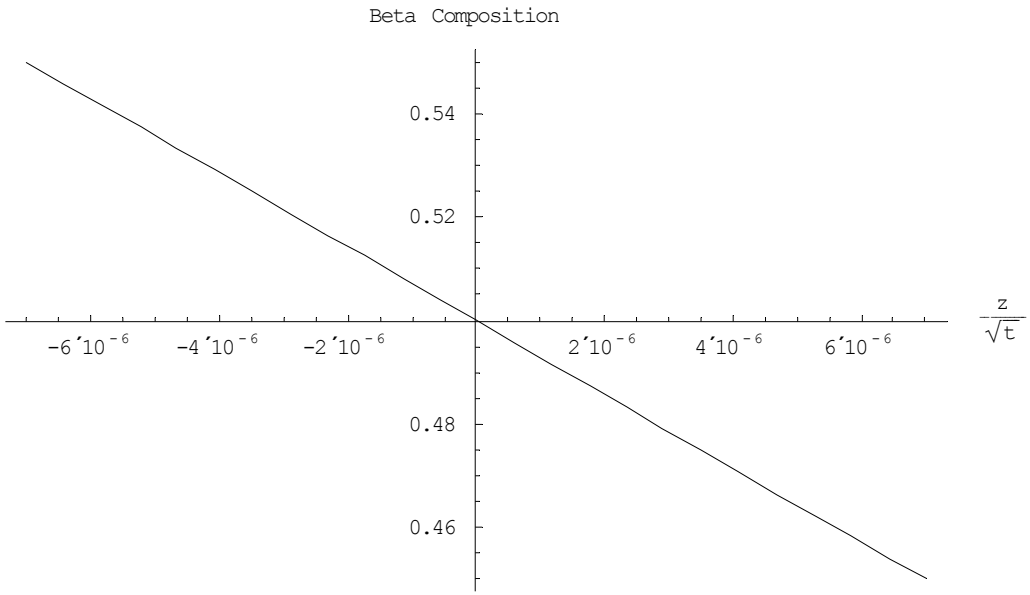
Check left and right compositions to see if you want to change leftside and rightside above.

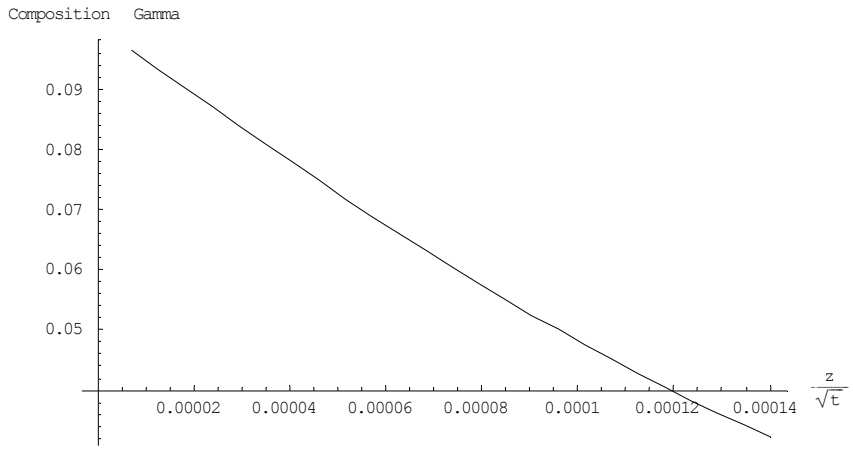
Also check that compositions at phase boundries are correct!

Calpha[kmin]	Calpha[k1]	Cbeta[k1]	Cbeta[k2]	Cgamma[k2]	Cgamma[kmax]
0.966505	0.9	0.55	0.45	0.096425	0.0322975

```
gy1 = Plot[Calpha[eta], {eta, kmin, k1}, AxesLabel → {z / Sqrt[t], Alpha Composition}];
gy2 = Plot[Cbeta[eta], {eta, k1, k2}, AxesLabel → {z / Sqrt[t], Beta Composition}];
gy3 = Plot[Cgamma[eta], {eta, k2, kmax}, AxesLabel → {z / Sqrt[t], Gamma Composition}];
```

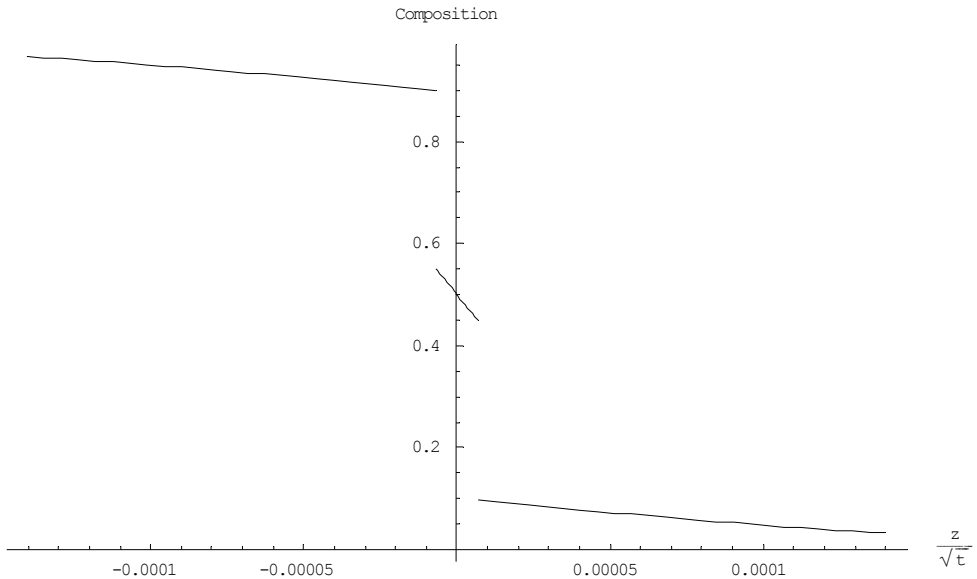






All plotted together

```
Show[{gy1, gy2, gy3}, AxesLabel → {z / Sqrt[t], Composition}]
```



- Graphics -

• Script to be posted to website
http://www.ctcms.nist.gov/~cecamp/workshop.html

• Script can be generalized to treat arbitrary number of layers.

• Script only treats local equilibrium (concentrations of each phase at the interfaces is fixed).

• Treats only situation where all interfaces start from the initial discontinuity in concentration at the same time.

• Solution using “FindRoot” fails when D-values differ greatly.

• Experience solving problem for dissolution of Cu in contact with liquid Sn and intermetallic growth of Cu_6Sn_5 by S. R. Coriell $\Rightarrow$ expand Exp[x]*Erf[x] for small x!