Electronic Annex

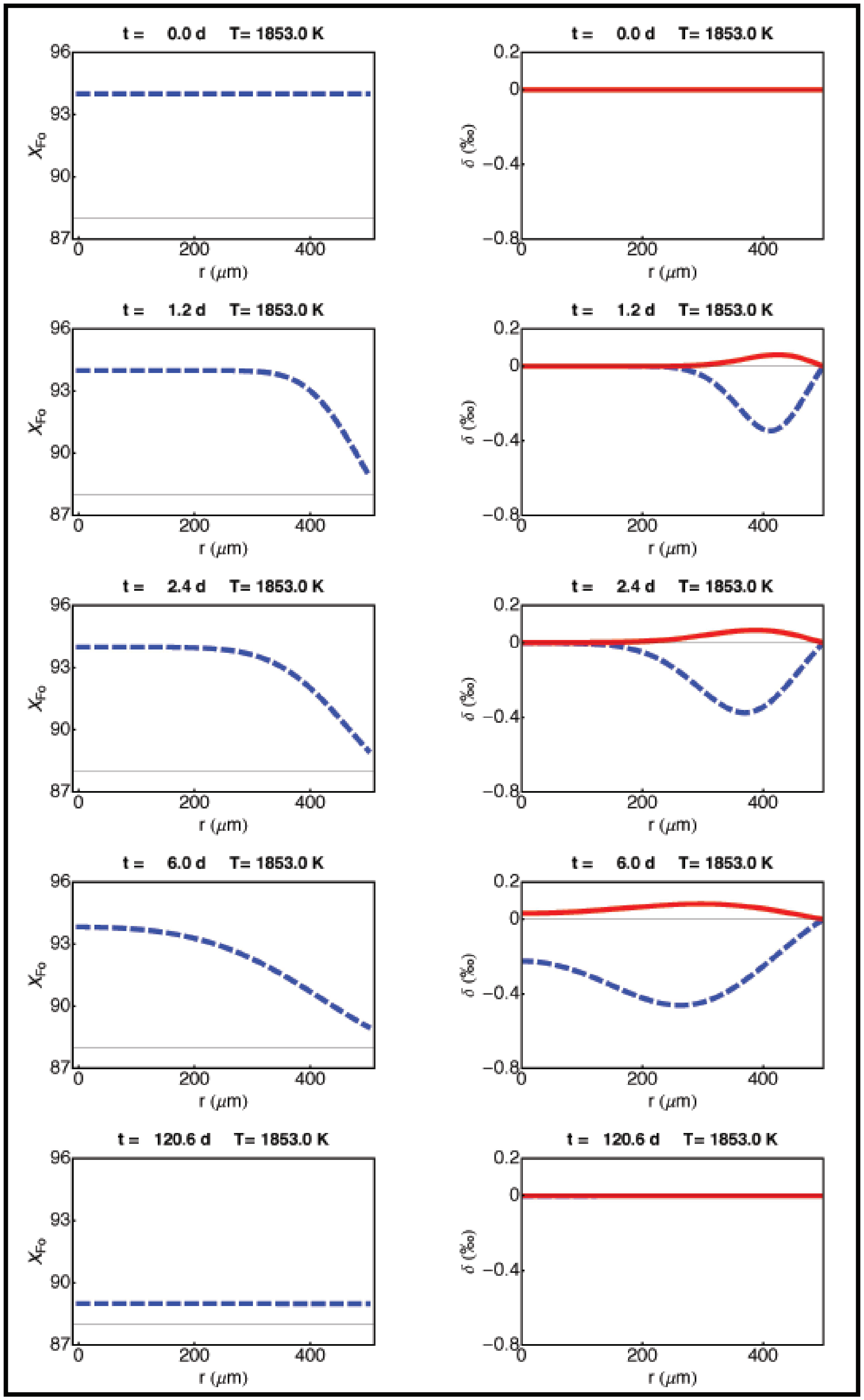
Figure. Test of the accuracy of the numerical solution to the problem of diffusive exchange of Mg and Fe in olivine. For an isothermal case with constant diffusion coefficient, the problem has a simple series solution (Crank, 1970) that can be compared with the result from the numerical calculation. The results of the two calculations (i.e., numerical and analytic) are strictly identical.

Code. Mathematica code used to solve the diffusion equations governing Mg-Fe exchange in olivine. The code can handle both linear and exponential cooling histories.

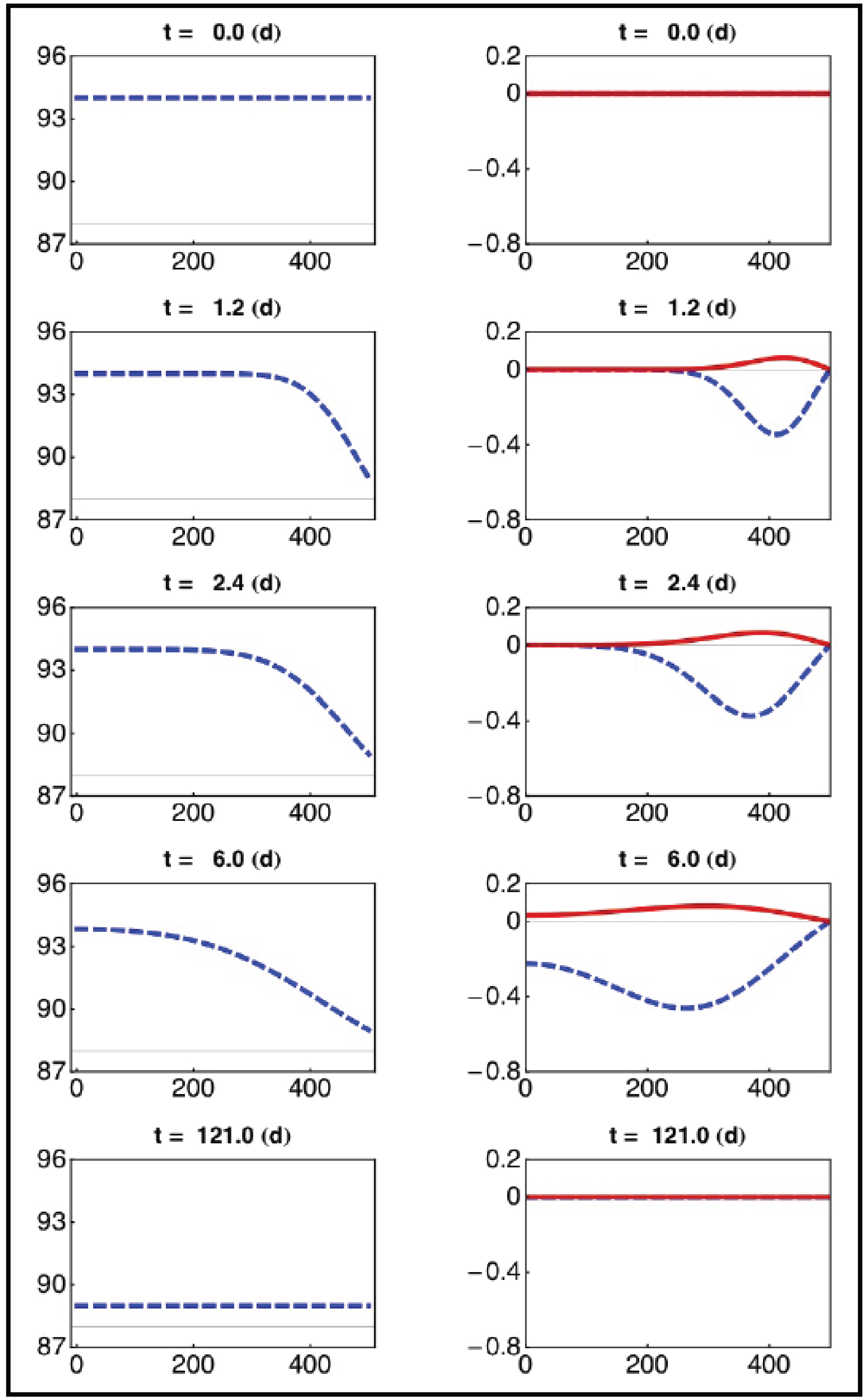
Animated gif. Animated version of Fig. 10 (opens in a web browser).

Test of the accuracy of the numerical solution

Numerical solution for isothermal case with $D=2.4\times10^{-14}\ m^2/s$



Series solution (Crank, 1970)



(* Fe-Mg isotope geospeedometry in olivine *)
(* constants *)
R = 8.314472; (* gas constant in J.K^{-1}.mol^{-1} *)

(* initialization *)
cooling = "linear"; (* "linear" T[t_]:=
If[T_0-t×c/(3600×24)>298.15,T_0-t×c/(3600×24),298.15] or "exponential" T[t_]:=
298.15+(T_0-298.15)×$e^{-t\times c/(3600\times 24)}$ *)
c = 30; (* cooling parameter in K.d^{-1} for linear cooling, in d^{-1} for exponential cooling *)
T_0 = 1853; (* initial temperature in K *)
β_{Mg} = 0.05; (* exponent in $D_2/D_1=(m_1/m_2)^{\beta}$ for Mg; Richter et al. 2009 *)
β_{Fe} = 0.05; (* exponent in $D_2/D_1=(m_1/m_2)^{\beta}$ for Fe; Richter et al. 2009 *)
ΔO_2 = 0; (* Log fO_2 relative to NNO buffer *)
P = 10^5; (* pressure in Pa *)
a = 300×10^{-6}; (* m, grain radius *)
X_{Fe1} = 0.11; (* surface Fe concentration, identical for 54 and 56 *)
X_{Mg1} = 0.89; (* surface Mg concentration, identical for 24 and 26 *)
X_{Fe0} = 0.06; (* initial Fe concentration, identical for 54 and 56 *)
X_{Mg0} = 0.94; (* initial Mg concentration, identical for 24 and 26 *)

(* definition of functions *)
T[t_] :=
If[cooling == "linear", If[T_0 - t × c / (3600 × 24) > 298.15, T_0 - t × c / (3600 × 24), 298.15],
298.15 + (T_0 - 298.15) × $e^{-t\times c/(3600\times 24)}$] (* temperature in K with t in s;
T[t_]:=If[T_0-t×c/(3600×24)>298.15,T_0-t×c/(3600×24),298.15] for linear cooling;
T[t_]:=298.15+(T_0-298.15)×$e^{-t\times c/(3600\times 24)}$ for exponential cooling *)

fO_2[T_, P_] := $101\,325 \times 10^{\Delta O_2} \times 10^{9.36-24\,930/T+0.046\times(P/10^5-1)/T}$
(* oxygen fugacity in Pa with T in K and P in Pa;
Huebner & Sato 1970, Chou 1987; Herd 2008 *)
d[T_, P_, X_, f_] := If[f > 10^{-10}, $10^{-9.21-\frac{201\,000+(P-10^5)\times7\times10^{-6}}{2.303\times R\times T}+\frac{1}{6}\mathrm{Log10}\left[\frac{f}{10^{-7}}\right]+3\,(X-0.1)}$, $10^{-8.91-\frac{220\,000+(P-10^5)\times7\times10^{-6}}{2.303\times R\times T}+3\,(X-0.1)}$]
(* Fe-Mg diffusion coefficient in olivine in m^2.s^{-1},
T in K, P pressure in Pa, f absolute oxygen fugacity in Pa,
X mole fraction of the fayalite component; Dohmen & Chakraborty 2007 *)

(* calculation of diffusion and cooling timescales *)
$\tau = \frac{a^2}{d[T_0,\ P,\ X_{Fe0},\ fO_2[T_0,\ P]]}$; (* diffusion timescale in s *)
Print["initial cooling rate for ", cooling,
" model: ", If[cooling == "linear", c, (T_0 - 298.15) × c], " K.d^{-1}"]
Print["cooling timescale: ", (T_0 - 1173.15) / If[cooling == "linear", c, (T_0 - 298.15) × c], " d"]
Print["diffusion timescale: ", τ / (3600 * 24), " d"];

```
(* f4, f6, m4, and m6 is the solution to the diffusion pde for 54Fe,
56Fe, 24Mg, and 26Mg respectively;
α is a coefficient that accounts for differences in diffusivities of isotopes *)
variable = {α → 1};
f4 = NDSolve[{D[X[t, r], t] ==
      1/(r + 0.000001 × 10^-6)^2 D[r^2 × α × d[T[t], P, X[t, r], fO2[T[t], P]] × D[X[t, r], r], r],
     X[0, r] == XFe0, X[t, a] == (1 / (1 + t)) × (XFe0 - XFe1) + XFe1,
     (D[X[t, r], r] /. r → 0) == 0} /. variable, X, {t, 0, 3 τ}, {r, 0, a},
   AccuracyGoal → MachinePrecision / 2, PrecisionGoal → MachinePrecision];
variable = {α → (54. / 56.)^βFe};
f6 = NDSolve[{D[X[t, r], t] ==
      1/(r + 0.000001 × 10^-6)^2 D[r^2 × α × d[T[t], P, X[t, r], fO2[T[t], P]] × D[X[t, r], r], r],
     X[0, r] == XFe0, X[t, a] == (1 / (1 + t)) × (XFe0 - XFe1) + XFe1,
     (D[X[t, r], r] /. r → 0) == 0} /. variable, X, {t, 0, 3 τ}, {r, 0, a},
   AccuracyGoal → MachinePrecision / 2, PrecisionGoal → MachinePrecision];
variable = {α → 1};
m4 = NDSolve[{D[X[t, r], t] ==
      1/(r + 0.000001 × 10^-6)^2 D[r^2 × α × d[T[t], P, (1 - X[t, r]), fO2[T[t], P]] × D[X[t, r], r], r],
     X[0, r] == XMg0, X[t, a] == (1 / (1 + t)) × (XMg0 - XMg1) + XMg1,
     (D[X[t, r], r] /. r → 0) == 0} /. variable, X, {t, 0, 3 τ}, {r, 0, a},
   AccuracyGoal → MachinePrecision / 2, PrecisionGoal → MachinePrecision];
variable = {α → (24. / 26.)^βMg};
m6 = NDSolve[{D[X[t, r], t] ==
      1/(r + 0.000001 × 10^-6)^2 D[r^2 × α × d[T[t], P, (1 - X[t, r]), fO2[T[t], P]] × D[X[t, r], r], r],
     X[0, r] == XMg0, X[t, a] == (1 / (1 + t)) × (XMg0 - XMg1) + XMg1,
     (D[X[t, r], r] /. r → 0) == 0} /. variable, X, {t, 0, 3 τ}, {r, 0, a},
   AccuracyGoal → MachinePrecision / 2, PrecisionGoal → MachinePrecision];
(* F4, F6, M4, M6 are the concentration profiles for 54Fe, 56Fe,
 24Mg, and 26Mg, respectively *)
F4[t_, r_] = X[t, r] /. f4[[1]];
```

```
F6[t_, r_] = X[t, r] /. f6[[1]];
M4[t_, r_] = X[t, r] /. m4[[1]];
M6[t_, r_] = X[t, r] /. m6[[1]];

δFe[t_, r_] = (F6[t, r] / F4[t, r] - 1) × 1000.;
δMg[t_, r_] = (M6[t, r] / M4[t, r] - 1) × 1000.;

(* δ56Fe of the bulk olivine grain *)
δFebulk[t_] := If[t == 0, 0, NIntegrate[δFe[t, r] × F4[t, r] × r^2, {r, 0, a}, MaxRecursion → 12] /
    NIntegrate[F4[t, r] × r^2, {r, 0, a}, MaxRecursion → 12]];
(* δ26Mg of the bulk olivine grain *)
δMgbulk[t_] := If[t == 0, 0, NIntegrate[δMg[t, r] × M4[t, r] × r^2, {r, 0, a}, MaxRecursion → 12] /
    NIntegrate[M4[t, r] × r^2, {r, 0, a}, MaxRecursion → 12]];


(* Ab plots the forsterite content of olivine
 as a function of radial position for any time g *)
Ab[g_] := Plot[M4[g, r] × 100, {r, 0, a}, AspectRatio → 1 / GoldenRatio,
  ImageSize -> Large, PlotRange → {88, 96}, Frame → True, FrameLabel → {"r (μm)", "X_Fo"},
  LabelStyle → Directive[FontFamily → "Helvetica", FontSize → 26],
  FrameTicks → {{{88, 92, 96}, None}, {{{0, 0}, {0.00015, 150}, {0.0003, 300}}, None}},
  FrameStyle → Thickness[0.005], PlotStyle → {{RGBColor[0, 0, 0],
     Dashing[{0.03}], Thickness[0.016]}, {RGBColor[0, 0, 0], Thickness[0.016]}},
  PlotLabel → Style[Row[{"t = ", PaddedForm[g / (3600 * 24), {5, 1}], " d     T=",
      PaddedForm[T[g], {5, 1}], " K"}], Bold, FontSize → 24]]

(* Is plots the isotopic compositions as a function of radial position for any time g *)
Is[g_] := Plot[{δFe[g, r], δMg[g, r]}, {r, 0, a}, AspectRatio → 1 / GoldenRatio,
  ImageSize -> Large, PlotRange → {{0, a}, {-0.9, 0.3}}, Frame → True, FrameLabel → {"r (μm)",
    Row[{Style["δ56Fe ", RGBColor[0, 0, 1]], Style["δ26Mg ", RGBColor[1, 0, 0]] , "(‰)"}]},
  LabelStyle → Directive[FontFamily → "Helvetica", FontSize → 26],
  FrameTicks → {{{-0.9, -0.6, -0.3, 0, 0.3}, None},
    {{{0, 0}, {0.00015, 150}, {0.0003, 300}}, None}}, FrameStyle → Thickness[0.005],
  PlotStyle → {{RGBColor[0, 0, 1], Dashing[{0.03}], Thickness[0.016]},
    {RGBColor[1, 0, 0], Thickness[0.016]}},
  PlotLabel → Style[Row[{"t = ", PaddedForm[g / (3600 * 24), {5, 1}],
      " d     T=", PaddedForm[T[g], {5, 1}], " K"}], Bold, FontSize → 24]]


(* Isis plots δ26Mg(r) vs δ56Fe(r) at each time *)
pcurve[g_] := ParametricPlot[{δFe[g, r], δMg[g, r]},
   {r, 0, a}, AspectRatio → 1 / GoldenRatio, ImageSize -> Large,
   PlotRange → {{-0.9, 0}, {0, 0.3}}, Frame → True, FrameLabel → {"δ56Fe (‰)", "δ26Mg (‰)"},
   LabelStyle → Directive[FontFamily → "Helvetica", FontSize → 26],
   FrameTicks → {{{0, 0.1, 0.2, 0.3}, None}, {{-0.9, -0.6, -0.3, 0}, None}},
   FrameStyle → Thickness[0.005], PlotStyle → {{RGBColor[0, 0, 0],
      Dashing[{0.03}], Thickness[0.016]}, {RGBColor[1, 0, 0], Thickness[0.016]}},
```

```
    PlotLabel → Style[Row[{"t = ", PaddedForm[g / (3600 * 24), {5, 1}], " d     T=",
        PaddedForm[T[g], {5, 1}], " K"}], Bold, FontSize → 24]];
bulkdot[g_] := ListPlot[{{δFebulk[g], δMgbulk[g]}}, AspectRatio → 1 / GoldenRatio,
    ImageSize -> Large, PlotRange → {{-0.9, 0}, {0, 0.3}},
    Frame → True, FrameLabel → {"δ56Fe (‰)", "δ26Mg (‰)"},
    LabelStyle → Directive[FontFamily → "Helvetica", FontSize → 26],
    FrameTicks → {{{0, 0.1, 0.2, 0.3}, None}, {{-0.9, -0.6, -0.3, 0}, None}},
    FrameStyle → Thickness[0.005], PlotLabel → Style[Row[{"t = ",
        PaddedForm[g / (3600 * 24), {5, 1}], " d     T=", PaddedForm[T[g], {5, 1}], " K"}],
      Bold, FontSize → 24], PlotStyle → Directive[PointSize[0.05], Red]];
bulkdotlabel[g_] = Graphics[Text[Style[Row[{"(", PaddedForm[δFebulk[g], {3, 2}], ",",
        PaddedForm[δMgbulk[g], {3, 2}], ")"}], Bold, FontSize → 24], {-0.17, +0.08}]];
Isis[g_] := Show[pcurve[g], bulkdot[g], bulkdotlabel[g]]


(* myfig contains a figure that is then exported to a pdf file *)
myfig = GraphicsGrid[{{Ab[0], Is[0], Isis[0]},
    {Ab[τ / 100], Is[τ / 100], Isis[τ / 100]}, {Ab[τ / 50], Is[τ / 50], Isis[τ / 50]},
    {Ab[τ / 20], Is[τ / 20], Isis[τ / 20]}, {Ab[3 * τ], Is[3 * τ], Isis[3 * τ]}}]
Export["/Users/dauphasu/Desktop/figure.pdf", myfig]

(* mymov contains the frames that are then exported to an animated GIF file *)
mymov = Table[{Ab[g], Is[g], Isis[g]}, {g, 0, τ / 5, τ / 500}];
Export["/Users/dauphasu/Desktop/animatedgif.gif", mymov]
```