References

[1] J. E. Wetzel, Sectorial covers for curves of constant length, Canad. Math. Bull. 16(1973) , 367-375.

[2] J. Gerriets and G. Poole, Convex regions which cover arcs of constant length, Amer. Math. Monthly 81 (1974) , 36-41.

[3] R. Norwood, G. Poole, and M. Laidacker, The worm problem of Leo Moser, Discrete Computational Geometry 7(1992) , 153-162.

[4] R. Norwood and G. Poole, An improved upperbound for Leo Moser's worm problem, Discrete Computational Geometry 29(2003), 409-417.

[5] W. Wong, Worm Problem, Acta Mathematica Sinica 49(2006), 835-847.

[6] P. Coulton and Y. Movshovich, Besicovitch Triangles Cover Unit Arcs, Geometriae Dedicata(2006), 138-147.

[7] J. E. Wetzel, The classical worm problem-a status report, Geombinatorics XV(2005), 34-42.

[8] M. S. Bazaraa , H. D. Sherali and C. M. Shetty, Nonlinear Programming Theory and Algorithms second edition (1993),408-489.

[9] S. Besicovitch, On arcs that cannot be covered by an open equilateral triangle of side 1, Math. Gaz. 49 (1965), 286-288.

[10] J. E. Wetzel, Fits and covers, Math. Magazine 76(2003), 349-363.

[11] J. E. Wetzel, The worm problem-a status report, Geometric Potpourri Seminar January 25 (2005) , 1-7

[12] W. Wichiramala, A smaller cover for convex unit arcs, East-West Journal of Mathematics 7(2005) :187-197

Appendix

Geometric figures and numerical minimization values in this research can be obtained from the following code by using Mathematica5.0

For equilateral triangle case

```
d[xys_] :=
 If[Length[xys] == 2, √(xys[[1, 1]] - xys[[2, 1]])^2 + (xys[[1, 2]] - xys[[2, 2]])^2,
  d[xys[[{1, 2}]]] + d[Rest[xys]]]

u[θ_] := {Cos[θ], Sin[θ]}; dd[xy_, θ_] := xy.u[θ]; sqrt3 = √3;

p1 = {x1, y1}; p1b = {x1b, y1b}; p2 = {x2, y2};
p3 = {x3, y3}; p4 = {x4, y4}; p5 = {x5, y5}; p6 = {x6, y6};

m = sqrt3; y1 = 0; y1b = 0; y2 = m x2; y3 = -m (x3 - 1); y4 = yA2; y5 = m (x5 - xA2) + yA2;
N[√(28 / 27)]
N[√(27 / 28)]

α = 60 °; β = 60 °

       0.5
AB = ------;
     Sin[α]
A1 = {0, 0}; C1 = {1, 0}; B1 = AB u[α];
A2 = {xA2, yA2}; C2 = {xA2 - 1, yA2}; B2 = A2 + AB u[180 ° + α];

u[θ_] = {Cos[θ], Sin[θ]};
Cross3[{x1_, y1_}, {x2_, y2_}] = x1 y2 - x2 y1;
OnRight[R_, P_, Q_] = Cross3[R - P, Q - P] > 0;
cond3 = OnRight[p3, C1, B1];
cond6 = dd[p6 - A2, 210 °] > √3 / 2
condt1 = y4 < dd[p5 - p2, α - 90 °];

draw[arc_, eval_] :=
 Show[Graphics[{{RGBColor[0, 1, 0], Line[{{0, 0}, {1, 0}, u[60 °], {0, 0}}]},
     {RGBColor[0, 0, 1], Line[{{xA2, yA2}, {xA2 - 1, yA2}, {xA2, yA2} - u[60 °],
         {xA2, yA2}}]}, {Thickness[.004], Line[arc]}} /. eval],
  AspectRatio → Automatic, PlotRange → {{-1, 2}, {-1, 2}}]
```

```
nm6 = {1, {x1 → 0.9999999955000765, x2 → 0.49999999589579347,
    x3 → 0.5000000005195876, x4 → 0.49999999529808775,
    x5 → 0.9999999948223148, x6 → 0.9999999906035069, xA2 → 1.4999999919558042,
    y6 → -3.422141170023855*^-9, yA2 → 0.8660253988581608}};
model6origin[nm0_, seq_, cond_, iter_, meth_] := (
  nm = NMinimize[{d[seq], cond3, cond6} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001},
     {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001}, {x4, x4 /. nm0[[2]],
      (x4 /. nm0[[2]]) + .000001}, {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001},
     {x6, x6 /. nm0[[2]], (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]],
      (y6 /. nm0[[2]]) + .000001}, {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001},
     {yA2, yA2 /. nm0[[2]], (yA2 /. nm0[[2]]) + .000001}},
    MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];)
model6[seq_, cond_, iter_] := model6origin[nm6, seq, cond, iter, Automatic]
model6again[seq_, cond_, iter_] := model6origin[nm, seq, cond, iter, Automatic]
model6DE[seq_, cond_, iter_] :=
 model6origin[nm6, seq, cond, iter, DifferentialEvolution]
model6againDE[seq_, cond_, iter_] :=
 model6origin[nm, seq, cond, iter, DifferentialEvolution]
model6NM[seq_, cond_, iter_] := model6origin[nm6, seq, cond, iter, NelderMead]
model6againNM[seq_, iter_] := model6origin[nm, seq, iter, NelderMead]

nm6e = {1, {x1 → 0.9999999955000765, x2 → 0.49999999589579347,
    x3 → 0.5000000005195876, x4 → 0.49999999529808775,
    x5 → 0.9999999948223148, x6 → 0.9999999906035069, xA2 → 1.4999999919558042,
    y6 → -3.422141170023855*^-9, yA2 → 0.8660253988581608}};
model6eorigin[nm0_, seq_, cond_, iter_, meth_] := (x4 = x1;
  nm = NMinimize[{d[seq], cond3, cond6} ⋃ cond,
    {{x1, x1 /. nm0[[2]], .000001 + x1 /. nm0[[2]]},
     {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001},
     {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001}, {x6, x6 /. nm0[[2]],
      (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]], (y6 /. nm0[[2]]) + .000001},
     {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001}, {yA2, yA2 /. nm0[[2]],
      (yA2 /. nm0[[2]]) + .000001}}, MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];, x4 =.)
model6e[seq_, cond_, iter_] := model6eorigin[nm6e, seq, cond, iter, Automatic]
model6eagain[seq_, cond_, iter_] := model6eorigin[nm, seq, cond, iter, Automatic]
model6eDE[seq_, cond_, iter_] :=
 model6eorigin[nm6e, seq, cond, iter, DifferentialEvolution]
model6eagainDE[seq_, cond_, iter_] :=
 model6eorigin[nm, seq, cond, iter, DifferentialEvolution]
model6eNM[seq_, cond_, iter_] := model6eorigin[nm6e, seq, cond, iter, NelderMead]
model6eagainNM[seq_, iter_] := model6eorigin[nm, seq, iter, NelderMead]
```

```
nm7 = {1, {x1 → 9.407630605393444*^-16, x1b → 0.9999997952025184,
    x2 → 2.839455236460388*^-15, x3 → 0.9999999366622955, x4 → 0.9999999366622953,
    x5 → 0.99999979520251, x6 → -2.8718226404576984*^-16, xA2 → 0.9999998585402131,
    y6 → 8.093426830227416*^-16, yA2 → 1.097041192074698*^-7}};
model7origin[nm0_, seq_, cond_, iter_, meth_] :=
 {nm = NMinimize[{d[seq], cond3, cond6, x1 < x4 < x1b} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001}, {x1b, x1b /. nm0[[2]],
      (x1b /. nm0[[2]]) + .000001}, {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001}, {x4, x4 /. nm0[[2]],
      (x4 /. nm0[[2]]) + .000001}, {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001},
     {x6, x6 /. nm0[[2]], (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]],
      (y6 /. nm0[[2]]) + .000001}, {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001},
     {yA2, yA2 /. nm0[[2]], (yA2 /. nm0[[2]]) + .000001}},
    MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];}
model7[seq_, cond, _iter_] := model7origin[nm7, seq, cond, iter, Automatic]
model7again[seq_, cond_, iter_] := model7origin[nm, seq, cond, iter, Automatic]
model7DE[seq_, cond_, iter_] :=
 model7origin[nm7, seq, cond, iter, DifferentialEvolution]
model7againDE[seq_, cond_, iter_] :=
 model7origin[nm, seq, cond, iter, DifferentialEvolution]
model7NM[seq_, cond_, iter_] := model7origin[nm7, seq, cond, iter, NelderMead]
model7againNM[seq_, iter_] := model7origin[nm, seq, iter, NelderMead]
```

For an isosceles righted angled triangle,

```
d[xys_] :=
 If[Length[xys] == 2, √((xys[[1, 1]] - xys[[2, 1]])^2 + (xys[[1, 2]] - xys[[2, 2]])^2),
  d[xys[[{1, 2}]]] + d[Rest[xys]]]

u[θ_] := {Cos[θ], Sin[θ]}; dd[xy_, θ_] := xy.u[θ]; sqrt3 = √3;

p1 = {x1, y1}; p1b = {x1b, y1b}; p2 = {x2, y2};
p3 = {x3, y3}; p4 = {x4, y4}; p5 = {x5, y5}; p6 = {x6, y6};

m = 1; y1 = 0; y1b = 0; y2 = m x2; y3 = -m (x3 - 1); y4 = yA2; y5 = m (x5 - xA2) + yA2;
```

$N\left[\frac{\sqrt{2}}{2}\right]$

0.707107

$\alpha = 45\,°;\ \beta = 45\,°$

45 °

$AB = \frac{0.5}{\mathrm{Sin}[\alpha]};$

```
A1 = {0, 0}; C1 = {1, 0}; B1 = AB u[α];
A2 = {xA2, yA2}; C2 = {xA2 - 1, yA2}; B2 = A2 + AB u[180 ° + α];

u[θ_] = {Cos[θ], Sin[θ]};
Cross3[{x1_, y1_}, {x2_, y2_}] = x1 y2 - x2 y1;
OnRight[R_, P_, Q_] = Cross3[R - P, Q - P] > 0;
cond3 = OnRight[p3, C1, B1];
cond6 = dd[p6 - A2, 225 °] > √2/2
condt1 = y4 < dd[p5 - p2, α - 90 °];
```

$-\frac{x6 - xA2}{\sqrt{2}} - \frac{y6 - yA2}{\sqrt{2}} > \frac{1}{\sqrt{2}}$

```
draw[arc_, eval_] :=
 Show[Graphics[{{RGBColor[0, 1, 0], Line[{{0, 0}, {1, 0}}, √2/2 u[45 °], {0, 0}}]},
    {RGBColor[0, 0, 1], Line[{{xA2, yA2}, {xA2 - 1, yA2}, {xA2, yA2} - √2/2 u[45 °],
       {xA2, yA2}}]}, {Thickness[.004], Line[arc]}} /. eval],
  AspectRatio → Automatic, PlotRange → {{-.5, 1.5}, {-.5, .6}}]

x1 → .5, x2 → .3, x3 → .7, x4 → .5, x5 → .7, x6 → .2, y6 → .1, xA2 → 1, yA2 → .4

nm6 = {1, {x1 → 0.9999518184201992, x2 → 0.5000302980145336, x3 → 0.9999809691454545,
    x4 → 0.5001300062320384, x5 → 0.9999721408382596, x6 → 0.5000580797878551,
    xA2 → 1.5000124273771929, y6 → 0.5000051033922595, yA2 → 0.5000541115504397}};
model6origin[nm0_, seq_, cond_, iter_, meth_] := {
  nm = NMinimize[{d[seq], cond3, cond6} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001},
     {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001}, {x4, x4 /. nm0[[2]],
      (x4 /. nm0[[2]]) + .000001}, {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001},
     {x6, x6 /. nm0[[2]], (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]],
      (y6 /. nm0[[2]]) + .000001}, {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001},
     {yA2, yA2 /. nm0[[2]], (yA2 /. nm0[[2]]) + .000001}},
    MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];}
model6[seq_, cond_, iter_] := model6origin[nm6, seq, cond, iter, Automatic]
model6again[seq_, cond_, iter_] := model6origin[nm, seq, cond, iter, Automatic]
model6DE[seq_, cond_, iter_] :=
 model6origin[nm6, seq, cond, iter, DifferentialEvolution]
model6againDE[seq_, cond_, iter_] :=
 model6origin[nm, seq, cond, iter, DifferentialEvolution]
model6NM[seq_, cond_, iter_] := model6origin[nm6, seq, cond, iter, NelderMead]
model6againNM[seq_, iter_] := model6origin[nm, seq, iter, NelderMead]
```

```
nm6e =
  {1, {x1 → 0.5482300155267756, x2 → 0.0021675964880911993, x3 → 0.999998147335638,
    x5 → 0.9999981473356386, x6 → 0.0021675964880911803, xA2 → 1.002165741960927,
    y6 → 0.00216759648809l243, yA2 → 0.00216944728964561l}};
model6eorigin[nm0_, seq_, cond_, iter_, meth_] := (x4 = x1;
  nm = NMinimize[{d[seq], cond3, cond6} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001},
     {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001},
     {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001}, {x6, x6 /. nm0[[2]],
      (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]], (y6 /. nm0[[2]]) + .000001},
     {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001}, {yA2, yA2 /. nm0[[2]],
      (yA2 /. nm0[[2]]) + .000001}}, MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]]; , x4 =.)
model6e[seq_, cond_, iter_] := model6eorigin[nm6e, seq, cond, iter, Automatic]
model6eagain[seq_, cond_, iter_] := model6eorigin[nm, seq, cond, iter, Automatic]
model6eDE[seq_, cond_, iter_] :=
 model6eorigin[nm6e, seq, cond, iter, DifferentialEvolution]
model6eagainDE[seq_, cond_, iter_] :=
 model6eorigin[nm, seq, cond, iter, DifferentialEvolution]
model6eNM[seq_, cond_, iter_] := model6eorigin[nm6e, seq, cond, iter, NelderMead]
model6eagainNM[seq_, iter_] := model6eorigin[nm, seq, iter, NelderMead]

nm7 = {1, {x1 → 0.36657353606884746, x1b → 0.9665387139253007,
    x2 → 0.09163584702319359, x3 → 0.9916353759462034, x4 → 0.6665765777629866,
    x5 → 0.9916353759462035, x6 → 0.09163584702319352, xA2 → 1.0832711475699537,
    y6 → 0.09163584702319352, yA2 → 0.10000039567754677}};
model7origin[nm0_, seq_, cond_, iter_, meth_] :=
 (nm = NMinimize[{d[seq], cond3, cond6, x1 < x4 < x1b} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001}, {x1b, x1b /. nm0[[2]],
      (x1b /. nm0[[2]]) + .000001}, {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001}, {x4, x4 /. nm0[[2]],
      (x4 /. nm0[[2]]) + .000001}, {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001},
     {x6, x6 /. nm0[[2]], (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]],
      (y6 /. nm0[[2]]) + .000001}, {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001},
     {yA2, yA2 /. nm0[[2]], (yA2 /. nm0[[2]]) + .000001}},
    MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];)
model7[seq_, cond, _iter_] := model7origin[nm7, seq, cond, iter, Automatic]
model7again[seq_, cond_, iter_] := model7origin[nm, seq, cond, iter, Automatic]
model7DE[seq_, cond_, iter_] :=
 model7origin[nm7, seq, cond, iter, DifferentialEvolution]
model7againDE[seq_, cond_, iter_] :=
 model7origin[nm, seq, cond, iter, DifferentialEvolution]
model7NM[seq_, cond_, iter_] := model7origin[nm7, seq, cond, iter, NelderMead]
model7againNM[seq_, cond_, iter_] := model7origin[nm, seq, cond, iter, NelderMead]
```

For a 30°- 60°-90° triangle,

```
d[xys_] := If[Length[xys] == 2,
  √((xys[[1, 1]] - xys[[2, 1]])^2 + (xys[[1, 2]] - xys[[2, 2]])^2),
  d[xys[[{1, 2}]]] + d[Rest[xys]]]

u[θ_] := {Cos[θ], Sin[θ]}; dd[xy_, θ_] := xy.u[θ]; sqrt3 = √3;

p1 = {x1, y1}; p1b = {x1b, y1b}; p2 = {x2, y2}; p3 = {x3, y3};
p4 = {x4, y4}; p5 = {x5, y5}; p6 = {x6, y6}; p7 = {x7, y7}; p8 {x8, y8};

m1 = Tan[π / 3]; m2 = Tan[π / 6]; y1 = 0; y1b = 0; y2 = m2 * x2; y3 = -m1 (x3 - 1);
y4 = yA2; y5 = m2 (x5 - xA2) + yA2; yA3 = -m1 (x3 - 1); yA4 = m1 (x6 - xA2 + 1) + yA2
N[1/√3]

√3 (1 + x6 - xA2) + yA2

0.57735

α = 30 °; β = 60 °

60 °

AB = 0.5/Sin[α];
A1 = {0, 0}; C1 = {1, 0}; B1 = AB u[α];
A2 = {xA2, yA2}; C2 = {xA2 - 1, yA2}; B2 = A2 + AB u[180 ° + α];
A3 = {xA3, yA3}; C3 = {xA3, yA3 - 1}; B3 = A3 + u[270 ° + α];
A4 = {xA4, yA4}; C3 = {xA4, yA4 + 1}; B4 = A4 + u[90 ° + α];

u[θ_] = {Cos[θ], Sin[θ]};
Cross3[{x1_, y1_}, {x2_, y2_}] = x1 y2 - x2 y1;
OnRight[R_, P_, Q_] = Cross3[R - P, Q - P] > 0;
cond3 = OnRight[p3, C1, B1];
cond6 = dd[p6 - A2, 210 °] > √3 / 2;
cond7 = x7 < xA3;
cond8 = x8 > xA4;
condt1 = y4 < dd[p5 - p2, α - 90 °];
```

```
draw[arc_, eval_] := Show[
  Graphics[{{RGBColor[0, 1, 0], Line[{{0, 0}, {1, 0}, {1, 0} + 1/2 u[120 °], {0, 0}}]},
     {RGBColor[0, 0, 1], Line[{{xA2, yA2}, {xA2 - 1, yA2}, {xA2, yA2} + √3/2 u[-150 °],
        {xA2, yA2}}]}, {Thickness[.004], Line[arc]}} /. eval],
  AspectRatio → Automatic, PlotRange → {{-.5, 2}, {-.5, .6}}]

nm6 = {1, {x1 → 0.20463559160489397, x2 → 0.2795381017954675,
    x3 → 0.838609432980824, x4 → 0.838609432980824, x5 → 0.763708005476749,
    x6 → 0.2046355916015553, xA2 → 1.043245022547358,
    y6 → 6.62816961450422*^-11, yA2 → 0.27953666193956267}};
model6origin[nm0_, seq_, cond_, iter_, meth_] := {
  nm = NMinimize[{d[seq], cond3, cond6} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001},
     {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001}, {x4, x4 /. nm0[[2]],
      (x4 /. nm0[[2]]) + .000001}, {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001},
     {x6, x6 /. nm0[[2]], (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]],
      (y6 /. nm0[[2]]) + .000001}, {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001},
     {yA2, yA2 /. nm0[[2]], (yA2 /. nm0[[2]]) + .000001}},
    MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];}
model6[seq_, cond_, iter_] := model6origin[nm6, seq, cond, iter, Automatic]
model6again[seq_, cond_, iter_] := model6origin[nm, seq, cond, iter, Automatic]
model6DE[seq_, cond_, iter_] :=
 model6origin[nm6, seq, cond, iter, DifferentialEvolution]
model6againDE[seq_, cond_, iter_] :=
 model6origin[nm, seq, cond, iter, DifferentialEvolution]
model6NM[seq_, cond_, iter_] := model6origin[nm6, seq, cond, iter, NelderMead]
model6againNM[seq_, iter_] := model6origin[nm, seq, iter, NelderMead]

nm6e =
  {1, {x1 → .5, x2 → .3, x3 → .7, x4 → .5, x5 → .7, x6 → .2, y6 → .1, xA2 → 1, yA2 → .4}};
model6eorigin[nm0_, seq_, cond_, iter_, meth_] := {x4 = x1;
  nm = NMinimize[{d[seq], cond3, cond6} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001},
     {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001},
     {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001}, {x6, x6 /. nm0[[2]],
      (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]], (y6 /. nm0[[2]]) + .000001},
     {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001}, {yA2, yA2 /. nm0[[2]],
      (yA2 /. nm0[[2]]) + .000001}}, MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]]; x4 =.}
model6e[seq_, cond_, iter_] := model6eorigin[nm6e, seq, cond, iter, Automatic]
model6eagain[seq_, cond_, iter_] := model6eorigin[nm, seq, cond, iter, Automatic]
model6eDE[seq_, cond_, iter_] :=
 model6eorigin[nm6e, seq, cond, iter, DifferentialEvolution]
model6eagainDE[seq_, cond_, iter_] :=
 model6eorigin[nm, seq, cond, iter, DifferentialEvolution]
model6eNM[seq_, cond_, iter_] := model6eorigin[nm6e, seq, cond, iter, NelderMead]
model6eagainNM[seq_, iter_] := model6eorigin[nm, seq, iter, NelderMead]
```

```
nm7 = {1, {x1 → 0.00015870181522523703, x1b → 0.998723667953158,
    x2 → 0.00012153457890932463, x3 → 1, x4 → 0.9987236557054678,
    x5 → 0.9999994803974714, x6 → 0.00015870181570898427, xA2 → 1.0001197031262972,
    y6 → -3.688713763988427*^-14, yA2 → 0.000067663021328 61847}};
model7origin[nm0_, seq_, cond_, iter_, meth_] :=
 {nm = NMinimize[{d[seq], cond3, cond6, x1 < x4 < x1b} ⋃ cond,
    {{x1, x1 /. nm0[[2]], (x1 /. nm0[[2]]) + .000001}, {x1b, x1b /. nm0[[2]],
      (x1b /. nm0[[2]]) + .000001}, {x2, x2 /. nm0[[2]], (x2 /. nm0[[2]]) + .000001},
     {x3, x3 /. nm0[[2]], (x3 /. nm0[[2]]) + .000001}, {x4, x4 /. nm0[[2]],
      (x4 /. nm0[[2]]) + .000001}, {x5, x5 /. nm0[[2]], (x5 /. nm0[[2]]) + .000001},
     {x6, x6 /. nm0[[2]], (x6 /. nm0[[2]]) + .000001}, {y6, y6 /. nm0[[2]],
      (y6 /. nm0[[2]]) + .000001}, {xA2, xA2 /. nm0[[2]], (xA2 /. nm0[[2]]) + .000001},
     {yA2, yA2 /. nm0[[2]], (yA2 /. nm0[[2]]) + .000001}},
    MaxIterations → iter, Method → meth],
  draw[seq, nm[[2]]];}
model7[seq_, cond, _iter_] := model7origin[nm7, seq, cond, iter, Automatic]
model7again[seq_, cond_, iter_] := model7origin[nm, seq, cond, iter, Automatic]
model7DE[seq_, cond_, iter_] :=
 model7origin[nm7, seq, cond, iter, DifferentialEvolution]
model7againDE[seq_, cond_, iter_] :=
 model7origin[nm, seq, cond, iter, DifferentialEvolution]
model7NM[seq_, cond_, iter_] := model7origin[nm7, seq, cond, iter, NelderMead]
model7againNM[seq_, iter_] := model7origin[nm, seq, iter, NelderMead]
```

VITA

Mr.Panuwat Tansatian was born on October 9th, 1982 in Nakornpathom Thailand. He graduated with a Bachelor of Science Degree in Mathematics from Mahidol University in 2005 (First Class Honor).