Bodensee: Version 1.0; Jacques Ambühl; 30 Juillet 2011

- **Directoire opérationnel**

```
SetDirectory["~/Mathematica/Bodensee/Operation"]
FileNames[] // TableForm
```

```
/Users/jacques/Mathematica/Bodensee/Operation

11072900_905.dat
BodenseeCoord.csv
Bodensee_OP.nb
Polaire.csv
```

Géométrie et navigation

- **Elements géométriques de base**

Convention: Longitude: Abscisse x ⟶ i ⟶ Θ
Latitude: Ordonnée y ⟶ j ⟶ Φ

```
fleche[{{x1_, y1_}, {x2_, y2_}}] := Module[{L, F},
   L = 0.8 ;
   F = 0.05 ;
   Line[{
     {x1, y1},
     {x1 + L (x2 - x1), y1 + L ( y2 - y1)},
     {x1 + L (x2 - x1) - F (y2 - y1), y1 + L ( y2 - y1) + F (x2 - x1)},
     {x2, y2},
     {x1 + L (x2 - x1) + F (y2 - y1), y1 + L ( y2 - y1) - F (x2 - x1)}, {x1 + L (x2 - x1), y1 + L ( y2 - y1)}
    }]]; (* fleche *)
```

```
Horaire[H_] := Module[{HH, MM},
   HH = Floor[H];
   MM = Round[60 (H - HH)];
   "[ " <> ToString[HH] <> "h " <> ToString[MM] <> "mm ]"];

DMS[x_] := Module[{D, M, , MS, Sec},
   D = Floor[x];
   MS = x - D;
   M = Floor[60 MS];
   Sec = Round[3600 MS - 60 M];
   ToString[M] <> "'"];
```

■ Navigation orthodromique

```
Track12[{x1_, y1_}, {x2_, y2_}] := Module[{θ1, ϕ1, θ2, ϕ2},
   {θ1, ϕ1, θ2, ϕ2} = π/180 {x1, y1, x2, y2};
   N[{Cos[ϕ2] Sin[θ2 - θ1], Cos[ϕ1] Sin[ϕ2] - Sin[ϕ1] Cos[ϕ2] Cos[θ1 - θ2]}]]; (* Track12 *)

Ortho[{x1_, y1_}, {x2_, y2_}] := Module[{θ1, ϕ1, θ2, ϕ2},
   {θ1, ϕ1, θ2, ϕ2} = π/180 {x1, y1, x2, y2};
   (* Km 6371 // NM 3437.75 *)
   360 60/(2 π) ArcCos[N[Cos[ϕ1] Cos[ϕ2] Cos[θ1 - θ2] + Sin[ϕ1] Sin[ϕ2]]]]; (* Ortho *)
```

Carte du Lac

```
Coast = Import["BodenseeCoord.csv"];
CoastLine = Graphics[{RGBColor[1, 0.5, 0], Line[Coast]}];
Marker = Graphics[{RGBColor[1, 0.5, 0], Point[Coast[[29]]]}];

TrameLat = Graphics[Table[{RGBColor[0.95, 0.95, 0.7], Line[{{9, 47.40 + 0.05 i}, {9.8, 47.40 + 0.05 i}}]}, {i, 0, 9}]];
TrameLong = Graphics[Table[{RGBColor[0.95, 0.95, 0.7], Line[{{9.0 + 0.05 i, 47.40}, {9.0 + 0.05 i, 47.85}}]}, {i, 0, 16}]];

TexteCoordonnées[La_, Lo_] := Module[{TexteLat, TexteLong},

   TexteLat = Graphics[Table[{Text[Style[DMS[47.40001 + 0.05 i], Small, Bold, Orange], {Lo, 47.40 + 0.05 i}]}, {i, 1, 9}]];
   TexteLong = Graphics[Table[{Text[Style[DMS[9.00001 + 0.05 i], Small, Bold, Orange], {9.0 + 0.05 i, La}]}, {i, 1, 16}]];
   {TexteLong, TexteLat}];(* TexteCoordonnées *)


Show[TrameLat, TrameLong, TexteCoordonnées[47.4, 9.0], CoastLine, Frame → True, FrameTicks → True]

(*
See=Import["BodenMap.jpg"];
Dimensions[See]
Show[See]
*)
```


51'
48'
45'
42'
39'
36'
33'
30'
27'
47.8
47.7
47.6
47.5
47.4
3'
6'
9'
12'
15'
18'
21'
24'
27'
30'
33'
36'
39'
42'
45'
48'
9.0
9.2
9.4
9.6
9.8

Polaire

- **Lecture de la table de base et construction de la fonction Yatch**

```
RawPol = Import["Polaire.csv"];

{DV, DA} = Dimensions[RawPol];
{MinW, MaxW, MinA, MaxA} = {RawPol[[2, 1]], RawPol[[DV, 1]], RawPol[[1, 2]], RawPol[[1, DA]]};
{Minr, Maxr} = {1, 7};
RawPolaire = Table[{{RawPol[[j, 1]], RawPol[[1, i]]}, RawPol[[j, i]]}, {j, 2, DV}, {i, 2, DA}] ;

MinimumSpeed = 0.001; (* KT *)
Voilier = Interpolation[Flatten[RawPolaire, 1]];
Navigateur = "Marco Stoll";
Yacht[W_, A_] :=
  If[(A < MinA) || (W < MinW) , MinimumSpeed,
   If[(W > MaxW) , Voilier[MaxW, A],
    Voilier[W, A]]];
```

Champ de vent

- **Lecture du Fichier COSMO-2**

```
MarcoField = Import["11072900_905.dat", "Table"];
Latitudesteps = 12;(* 12 steps in Latitude in Marco's File *)
EchéancesPrévi = 24; (* 24 échéances de prévision *)
LastTabWnd = 337; (* Tête de la dernière table de vent *)
{Nwest, Nnorth} = {First[Dimensions[MarcoField[[1]]]] - 1, Latitudesteps}
{Mmax, Nmax} = {Nwest, Nnorth}
```

```
{31, 12}
```

```
{31, 12}
```

```
MarcoField[[1, 1]]
```

```
2011JUL2900
```

Coordonnées de base du champ de vent

```
{δlat, δlong} = {MarcoField[[1, 4]] - MarcoField[[1, 3]], - MarcoField[[4, 1]] + MarcoField[[3, 1]]};
{LongMin, LatMax} = {MarcoField[[1, 2]], MarcoField[[2, 1]]};
{LongMax, LatMin} = {Last[MarcoField[[1]]], MarcoField[[Latitudesteps + 1, 1]]};

Coordinates = Table[{LongMin + δlong i, LatMax - δlat j}, {j, 0, Nnorth - 1}, {i, 0, Nwest - 1}];
```

Construction du champ de vent (20 pas dans le fichier entre chaque échéance !!)

```
WindLenghtFactor = 0.005;

WindField[ech_] := Module[{ForecastStep, W1s, W1e, W, TZero},
    ForecastStep = 1 + (Latitudesteps + 2) ech;
    {W1s, W1e} = {ForecastStep, ForecastStep + Latitudesteps};
    W = Table[MarcoField[[i]], {i, W1s, W1e}];
    {W[[1, 1]] , Table[3600/1852 {W[[i, j - 1]], W[[i, j]]}, {i, 2, Nnorth + 1}, {j, 3, 2 Nwest, 2}]}
  ];
```

Préparation du Graphe

- **Structure et orientation du Graphe**

```
GrapheGeneration[{θ1_, ϕ1_}, {θ2_, ϕ2_}, {m_, n_, ρ_}] := Module[{Bornes, Coord, MiLo, MaLo, MiLa, MaLa, CoordLong, CoordLat, e, c, s, M},
  e = Sqrt[(θ2 - θ1)^2 + (ϕ2 - ϕ1)^2];
  {c, s} = {(θ2 - θ1)/e, (ϕ2 - ϕ1)/e};
  M := {{c, -s}, {+s, c}};

  Coord = Table[N[M.{e/m i, ρ e/m j} + {θ1, ϕ1}], {i, 0, m}, {j, -n, n}];

  CoordLong = Table[Coord[[k, l, 1]], {k, 1, m + 1}, {l, 1, 2 n + 1}];
  CoordLat = Table[Coord[[k, l, 2]], {k, 1, m + 1}, {l, 1, 2 n + 1}];
  {MiLo, MaLo, MiLa, MaLa} = {Min[CoordLong], Max[CoordLong], Min[CoordLat], Max[CoordLat]};
  Bornes = {{MiLo - (MaLo - MiLo)/20, MaLo + (MaLo - MiLo)/20}, {MiLa - (MaLa - MiLa)/20, MaLa + (MaLa - MiLa)/20}};
  {Coord, Bornes}
]; (* GrapheGeneration *)
```

Interpolation des vitesses de vent aux sommets du graphe

```
BuildWindTable[Ech_] := Module[{Wind, WindInterpolation},

  WindInterpolation[{x_, y_}] := Module[{WindInterpolationFull, Vide, A, MinA, Refi, Refj, Refx, Refy},

    WindInterpolationFull[{xi_, yi_}] :=
     Module[{SquareCoordinates, Void, Csw, Cse, Cne, Cnw, CordSW, CordSE,
       CordNW, CordNE, αZonal, αMeridional, WndSW, WndSE, WndNE, WndNW, WndW, WndE, Boat, grapheWindBoat},
```

```
SquareCoordinates[{xa_, ya_}] := Module[{se, sw, nw, ne, type},
  If[((xa ≥ Refx) && (ya ≥ Refy)),
   {type, sw, se, ne, nw} = {"a", {Refi, Refj}, {Refi, Refj + 1}, {Refi - 1, Refj + 1}, {Refi - 1, Refj}},
   If[((xa ≥ Refx) && (ya ≤ Refy)), {type, sw, se, ne, nw} = {"b", {Refi + 1, Refj}, {Refi + 1, Refj + 1}, {Refi, Refj + 1}, {Refi, Refj}},
    If[((xa ≤ Refx) && (ya ≤ Refy)), {type, sw, se, ne, nw} =
      {"c", {Refi + 1, Refj - 1}, {Refi + 1, Refj}, {Refi, Refj}, {Refi, Refj - 1}}, If[((xa ≤ Refx) && (ya ≥ Refy)),
      {type, sw, se, ne, nw} = {"d", {Refi, Refj - 1}, {Refi, Refj}, {Refi - 1, Refj}, {Refi - 1, Refj - 1}}]]]]
 ]; (* End Module SquareCoordinates *)

{Void, Csw, Cse, Cne, Cnw} = SquareCoordinates[{xi, yi}];

CordSW = Coordinates[[First[Csw], Last[Csw]]]; (* turquoise *)
CordSE = Coordinates[[First[Cse], Last[Cse]]]; (* bleu *)
CordNE = Coordinates[[First[Cne], Last[Cne]]]; (* vert *)
CordNW = Coordinates[[First[Cnw], Last[Cnw]]]; (* rouge *)

{αZonal, αMeridional} = {(xi - First[CordSW])/(First[CordSE] - First[CordSW]), (yi - Last[CordSE])/(Last[CordNE] - Last[CordSE])};

WndSW = Wind[[First[Csw], Last[Csw]]];
WndSE = Wind[[First[Cse], Last[Cse]]];
WndNE = Wind[[First[Cne], Last[Cne]]];
WndNW = Wind[[First[Cnw], Last[Cnw]]];

WndW = αMeridional WndNW + (1 - αMeridional) WndSW;
WndE = αMeridional WndNE + (1 - αMeridional) WndSE;

αZonal WndE + (1 - αZonal) WndW]; (* WindInterpolationFull *)

A = Table[N[Ortho[{x, y}, Coordinates[[i, j]]]], {i, 1, Nmax}, {j, 1, Mmax}];

MinA = Min[A];
{Refi, Refj} = First[Position[A, MinA]];
{Refx, Refy} = Coordinates[[Refi, Refj]];

If[MinA < 0.1 δlat, Wind[[Refi, Refj]], WindInterpolationFull[{x, y}]]
]; (* End Module WindInterpolation *)

Wind = Last[WindField[Ech]];

Table[If[InTruth[[i, j]], WindInterpolation[Locations[[i, j]]], {MinimumSpeed, MinimumSpeed}], {i, 1, Maxi}, {j, 1, Maxj}]
```

```
]; (* BuildWindTable *)
```

Présentation Graphique

```
Arbre := Module[{G4, G5},
   G4 = Table[Graphics[{RGBColor[1, 0.7, 0.3], Point[Locations[[i, j]]]}], {i, 1, Maxi}, {j, 1, Maxj}];
   G5 = Table[Graphics[{RGBColor[0.9, 0.7, 0.7], Line[{
         Locations[[First[Analyse[[i, 1, 1]]], Last[Analyse[[i, 1, 1]]]]],
         Locations[[First[Analyse[[i, 1, 2]]], Last[Analyse[[i, 1, 2]]]]]}]}], {i, 1, DimD}];
   {(*G4,*)G5}];(* Arbre *)

Isochr[Z_] := Module[{Selec, DimS, Iso, ϵ},
   ϵ = 0.01;
   Selec = Select[Analyse, (#[[2]] < Z) && (Z < #[[3]]) &];
   DimS = First[Dimensions[Selec]];

   Iso[z_, i_] := Module[{α, S1, S2},
     α = (z - Selec[[i, 2]]) / (Selec[[i, 3]] - Selec[[i, 2]]);

     S1 = Locations[[First[Selec[[i, 1, 1]]], Last[Selec[[i, 1, 1]]]]];
     S2 = Locations[[First[Selec[[i, 1, 2]]], Last[Selec[[i, 1, 2]]]]];

     (1 - α) S1 + α S2]; (* Iso *)

   {Table[Graphics[{RGBColor[1, 0, 0], Point[Iso[Z, i]]}], {i, 1, DimS}], (* Graphics[
     Text[Style[Z,Small,Bold, Red],Iso[Z,1]-{ϵ,ϵ}]], *)Graphics[Text[Style[Z, Small, Bold, Red], Iso[Z, 1(* DimS *)] - {ϵ, ϵ}]]}
  ]; (* Isochr *)
```

```
GrapheFieldStatique[ech_] :=
  Module[{ Ech, W3, T1, T2, VerifTime, grapheGrid, grapheWind1, grapheWind2, TitreSimple},

   TitreSimple := Module[{TCoord, Texte0, Texte1, Texte2, Texte3, Texte4, Texte5, Texte6, Echeances, DimHeading, S, ϵ},
     ϵ = 0.015;

     TCoord = BormesLocations;
     Echeances = ToString[CosmoStart] <> " - " <> ToString[CosmoEnd];

     Texte1 = Graphics[{Text[Style["Prévision: COSMO-2", Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 5 ϵ}]}];
     Texte2 =
      Graphics[{Text[Style["Echéances: " <> Echeances <> " UTC", Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 4 ϵ}]}];
     Texte3 = Graphics[{Text[Style["Date: " <> ForecastDay, Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 3 ϵ}]}];;
     Texte4 = Graphics[{Text[Style["©Routage 2011: J. Ambühl", Small, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 0.45 ϵ}]}];

     {TexteCoordonnées[TCoord[[2, 1]] + ϵ, TCoord[[1, 1]] + ϵ], Texte1, Texte2, Texte3, Texte4}];


   Ech = Round[ech] - CosmoStart;
   W3 = WindField[Ech];
   VerifTime = Graphics[{Text[Style[First[W3], Medium, Bold, Blue], {9.68, 47.77}]}];
   grapheGrid =
    Graphics[{PointSize[0.003], RGBColor[0.4, 0.4, 0.7], Table[Point[Coordinates[[j, i]]], {j, Nmax, 1, -1}, {i, 1, Mmax}]}];

   T1 = Table[{Coordinates[[i, j]], Coordinates[[i, j]] + WindLenghtFactor Last[W3][[i, j]]}, {i, 1, Nnorth - 1}, {j, 1, Nwest - 1}];
   grapheWind1 = Graphics[{RGBColor[0.2, 0.6, 1], Thickness[0.001], Table[fleche[T1[[i, j]]], {i, 1, Nmax - 1}, {j, 1, Mmax - 1}]}];

   T2 = Table[{Locations[[i, j]], Locations[[i, j]] + WindLenghtFactor ContinuousWind[i, j, ech]}, {i, 1, Maxi}, {j, 1, Maxj}];
   grapheWind2 = Graphics[{RGBColor[0.4, 0.2, 1], Thickness[0.001], Table[fleche[T2[[i, j]]], {i, 1, Maxi}, {j, 1, Maxj}]}];

   Show[TrameLong, TrameLat, CoastLine, TexteCoordonnées[47.4, 9.0],
    grapheGrid, grapheWind1, grapheWind2, VerifTime, TitreSimple,
    (*Arbre,Isochr[ech],*) Frame → True, PlotRange → BormesLocations]
  ]; (* GraphFieldStatique *)
```

```
Titres := Module[{TCoord, Texte0, Texte1, Texte2, Texte3, Texte4, Texte5, Texte6, Echeances, DimHeading, S, ϵ},
   ϵ = 0.015;

   TCoord = BormesLocations;
   Echeances = ToString[CosmoStart] <> " - " <> ToString[CosmoEnd];
   Texte0 = Graphics[{Text[Style["Voilier: " <> Navigateur, Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 6 ϵ}]}];
   Texte1 = Graphics[{Text[Style["Prévision: COSMO-2", Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 5 ϵ}]}];
   Texte2 = Graphics[{Text[Style["Echéances: " <> Echeances <> " UTC", Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 4 ϵ}]}];
   Texte3 = Graphics[{Text[Style["Date: " <> ForecastDay, Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 3 ϵ}]}];;
   Texte4 = Graphics[{Text[Style["©Routage 2011: J. Ambühl", Small, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 0.45 ϵ}]}];

   DimHeading = First[Dimensions[Heading]];
   Texte5 = Graphics[Table[{Text[Style[Heading[[S, 1, 2]], Small, Bold, Blue], {ϵ, ϵ} + Heading[[S, 1, 1]]]}, {S, 1, DimHeading}]];
   Texte6 =
    Graphics[{Text[Style["Départ [GMT]: " <> Horaire[DepartureTime], Medium, Bold, Blue], {TCoord[[1, 1]] + 5 ϵ, TCoord[[2, 1]] + 2 ϵ}]}];

   {TexteCoordonnées[TCoord[[2, 1]] + ϵ, TCoord[[1, 1]] + ϵ], Texte0, Texte1, Texte2, Texte3, Texte4, Texte5, Texte6}];
```

```
GrapheFieldDynamique[IsoStart_] := Module[{Ech, T3, T4, T5, grapheWind3, grapheWind4, grapheDépart, grapheArrivée, grapheRoute, ϵ},

   ϵ = 0.002;
   T3 = Table[{
      Locations[[First[Analyse[[i, 1, 1]]], Last[Analyse[[i, 1, 1]]]]],
      Locations[[First[Analyse[[i, 1, 1]]], Last[Analyse[[i, 1, 1]]]] +
       WindLenghtFactor ContinuousWind[First[Analyse[[i, 1, 1]]], Last[Analyse[[i, 1, 1]]], Analyse[[i, 2]]]}, {i, 1, DimD}];

   grapheWind3 = Graphics[{RGBColor[0.7, 0.7, 1], Thickness[0.001], Table[fleche[T3[[i]]], {i, 1, DimD}]}];

   T4 = Table[{
      Locations[[First[Analyse[[i, 1, 2]]], Last[Analyse[[i, 1, 2]]]]],
      Locations[[First[Analyse[[i, 1, 2]]], Last[Analyse[[i, 1, 2]]]] +
       WindLenghtFactor ContinuousWind[First[Analyse[[i, 1, 2]]], Last[Analyse[[i, 1, 2]]], Analyse[[i, 3]]]}, {i, 1, DimD}];

   grapheWind4 = Graphics[{RGBColor[0.7, 0.7, 1], Thickness[0.001], Table[fleche[T4[[i]]], {i, 1, DimD}]}];

   T5 =
    Table[{Locations[[Route[[i + 1, 1]], Route[[i + 1, 2]]]], Locations[[Route[[i, 1]], Route[[i, 2]]]]}, {i, 1, GraphAttributes[[1]]}];

   grapheRoute = Graphics[{RGBColor[0.0, 0.0, 1], Thickness[0.003], Table[fleche[T5[[j]]], {j, 1, GraphAttributes[[1]]}]}];

   grapheDépart = Graphics[{Blue, Rectangle[DepartureLocation - {ϵ, ϵ}, DepartureLocation + {ϵ, ϵ}]}];
   grapheArrivée = Graphics[{Blue, Rectangle[ArrivalLocation - {ϵ, ϵ}, ArrivalLocation + {ϵ, ϵ}]}];

   Show[TrameLong, TrameLat, CoastLine, Arbre,
        grapheWind3, grapheWind4, grapheRoute, grapheDépart, grapheArrivée,
        Isochr[Ceiling[IsoStart] + 0],
        Isochr[Ceiling[IsoStart] + 1],
        Isochr[Ceiling[IsoStart] + 2],
        Isochr[Ceiling[IsoStart] + 3],
        Isochr[Ceiling[IsoStart] + 4],
    Titres, PlotRange → BormesLocations, Frame → True, FrameTicks -> None]

  ];(* GraphFieldDynamique *)
```

Algorithme de décision: programmation dynamique

Préparation

```
TimeTuning := Module[{FDay, HS, HE, DS, DE, CE},

    (* Time settings *)

    FDay = StringDrop[MarcoField[[1, 1]], -2];

    HS = ToExpression[StringDrop[MarcoField[[1, 1]], 9]]; (* Tête de la première table de vents *)
    HE = ToExpression[StringDrop[MarcoField[[LastTabWnd, 1]], 9]]; (* Tête de la dernière table de vents *)

    DS = ToExpression[StringTake[MarcoField[[1, 1]], {8, 9}]];
    DE = ToExpression[StringTake[MarcoField[[267, 1]], {8, 9}]];

    {FDay, HS, HS + EchéancesPrévi}]; (* 24 Echéances de prévision, Timestep 1 h. *)


TimeTuning
```

```
{2011JUL29, 0, 24}
```

```
ContinuousWind[i_, j_, t_] := Module[{Ech, δEch, WindAnte, WindPost},
    If[(t < CosmoStart) || (CosmoEnd < t), {MinimumSpeed, MinimumSpeed}, (* vérifier !! *)
     Ech = IntegerPart[t] - CosmoStart;
     δEch = t - (Ech + CosmoStart);
     WindAnte = WindinTime[[Ech + 1, i, j]];
     WindPost = WindinTime[[Ech + 2, i, j]];
     (1 - δEch) WindAnte + δEch WindPost]
   ];
```

Calcul direct des options possibles

```
ProgrammationDynamique[Start_, Time_, Spread_] := Module[{Graphe, DimGraphe, Duration, ProgrDyn, G0, G1, DimG1, G2, G3, G4},

    Duration[i1_, j1_, i2_, j2_] :=
     Module[{t1, t2, Lx, Ly, Compute, FixPoint, Route}, {Lx, Ly} = Track12[Locations[[i1, j1]], Locations[[i2, j2]]];

      {t1, t2} = {Graphe[[i1, j1, 3]], Graphe[[i2, j2, 3]]};

      Compute[tt1_, tt2_] := Module[{Wu, Wv, WSpeedArrival, AllureArrival, WSpeedStart, AllureStart, YArrival, YStart, Leg},
```

```
{Wu, Wv} = ContinuousWind[i1, j1, tt1];
WSpeedStart = √(Wu^2 + Wv^2);
AllureStart = 180 (1 - 1/π ArcCos[{Wu, Wv}.{Lx, Ly} / (WSpeedStart √(Lx^2 + Ly^2))]);
YStart = Yacht[WSpeedStart, AllureStart];

{Wu, Wv} = ContinuousWind[i2, j2, tt2];
WSpeedArrival = √(Wu^2 + Wv^2);
AllureArrival = 180 (1 - 1/π ArcCos[{Wu, Wv}.{Lx, Ly} / (WSpeedArrival √(Lx^2 + Ly^2))]);
YArrival = Yacht[WSpeedArrival, AllureArrival];

Leg = {
   (WSpeedStart + WSpeedArrival) / 2,
   (AllureStart + AllureArrival) / 2,
   (YStart + YArrival) / 2};

(*
If[RandomInteger[{1,10000}]≥9950,
 Print[" "    Cap du Vent: ",(CapVentStart+CapVentArrival)/2,
            "    Vitesse du vent: ",(WSpeedStart+WSpeedArrival)/2,
              "    Allure: ",(AllureStart+AllureArrival)/2,
                "    Vitesse du voilier: ",(YStart+YArrival)/2]];
          *)

  {2 Ortho[Locations[[i1, j1]], Locations[[i2, j2]]] / (YStart + YArrival), Leg}
 ]; (* Compute *)

(*
FixPoint[t_]:=Module[{D0,D1,DD1,D2,ϵ},
  D0=Compute[t,t];
```

```
    Do[{D1=Compute[t,t+D0];
      ϵ= 0.01D1;
      DD1= (Compute[t,t+D1+ϵ]-Compute[t,t+D1-ϵ])/(2ϵ);
      D2=D0-(D1-D0)/(DD1-1);
      D0=D2},
     {i,1,2}];D2];

  FixPoint[t1]
  *)

  Compute[t1, t1 + First[Compute[t1, t1 + First[Compute[t1, t1]] ]] ]

 ]; (* Duration *)

ProgrDyn[r_] := Module[{Etabli, Posi, Wahl, MinEtabli},

  Do[
   Wahl = Table[Duration[r - 1, j, r, k], {j, 1, Maxj}];
   Etabli = Table[Graphe[[r - 1, j, 3]] + First[Wahl[[j]]], {j, 1, Maxj}];
   MinEtabli = Min[Etabli];
   {{Posi}} = Position[Etabli, MinEtabli];

   If[Abs[k - Posi] ≤ Spread, Graphe[[r, k]] =
     {{{r - 1, Posi}, {r, k}}, Graphe[[r - 1, Posi, 3]], MinEtabli, Last[Wahl[[Posi]]] }],
   {k, 1, Maxj}]

 ]; (* ProgrDyn *)

Graphe = Table[{0, 0, 10^6, 10^6}, {i, 1, Maxi}, {j, 1, Maxj}];
Graphe[[1, Start]] = {1, 0, Time};

Do[ProgrDyn[r], {r, 2, Maxi}];

G0 = Table[Drop[Graphe[[i, j]], -1 ], {i, 1, Maxi}, {j, 1, Maxj}];
G1 = Select[Flatten[Transpose[G0], 1], (Last[#] < 1000000) && (Last[#] > Time) &];
DimG1 = First[Dimensions[G1]];

G2 = Table[First[G1[[i]]], {i, 1, DimG1}];
G3 = Table[{{First[G2[[i]]][[1]], First[G2[[i]]][[2]]}, {Last[G2[[i]]][[1]], Last[G2[[i]]][[2]]}}, {i, 1, DimG1}];
G4 = Table[{G3[[i]], G1[[i, 2]], G1[[i, 3]]}, {i, 1, DimG1}];
```

```
    {Graphe, G4}

  ]; (* Programmation Dynamique *)
```

Calcul rétrograde de l'option choisie et rédaction du logbook

```
WohinDesWegs := Module[{Vorgänger, Weg, nn, LB},
   Vorgänger[{i_, j_}] := Réseau[[i, j, 1, 1]];
   Weg = {ArrivalPoint};
   nn = Maxi - 1; While[nn ≥ 1, Weg = Append[Weg, Vorgänger[Last[Weg]]]; nn--];
   LB = Table[{i, Réseau[[Weg[[i, 1]], Weg[[i, 2]]]]}, {i, 1, Maxi - 1}];
   {Weg, LB}
  ];(* CalculRoute *)

LogBook := Module[{LB},

   LB[S_] := Module[{ALong, ALat, BLong, BLat, LocLong, LocLat, ButLong, ButLat, DMS, Komp, Kompass, Heure, Flottheit},

     DMS[x_] := Module[{D, M, , MS, Sec},
       D = Floor[x];
       MS = x - D;
       M = Floor[60 MS];
       Sec = Round[3600 MS - 60 M];
       "[ " <> ToString[D] <> "° " <> ToString[M] <> "m " <> ToString[Sec] <> "s ]"];

     Heure = Logbuch[[S, 2, 2]];
     {ALong, ALat} = Logbuch[[S, 2, 1, 1]];
     {BLong, BLat} = Logbuch[[S, 2, 1, 2]];
     {LocLong, LocLat} = Locations[[ALong, ALat]];
     {ButLong, ButLat} = Locations[[BLong, BLat]];
     Komp = ToString[Round[GeoDirection[{LocLat, LocLong}, {ButLat, ButLong}]]];
     Kompass = "[ Kompass: " <> Komp <> "° ]";
     Flottheit = "[ Speed: " <> ToString[Logbuch[[S, 2, 4, 3]]] <> " Kt ]";
     {{{LocLong, LocLat}, Komp}, {Horaire[Heure], DMS[LocLat], DMS[LocLong], Kompass, Flottheit}}
    ]; (* LB *)

   {
    Table[{First[LB[Maxi - i]]}, {i, 1, Maxi - 1}], Table[ {i, Last[LB[Maxi - i]]}, {i, 1, Maxi - 1}]}

  ];(* LogBook *)
```

Action

- **Initialisation et construction du graphe**

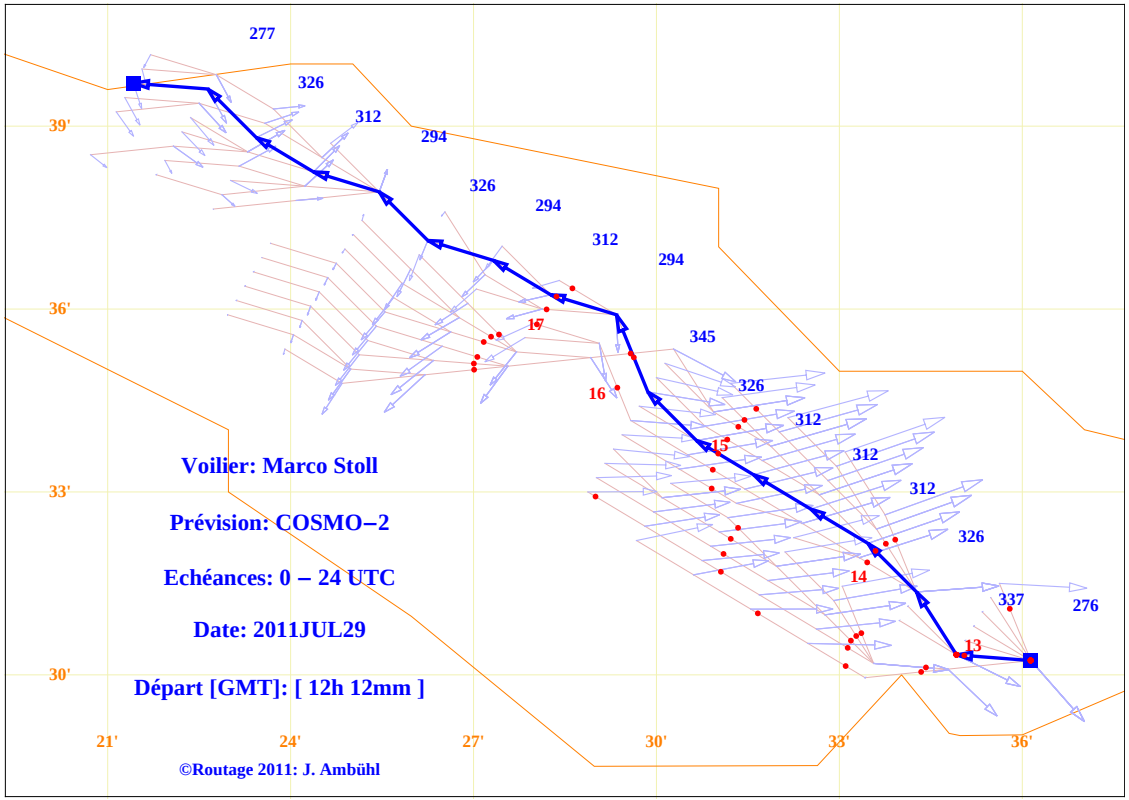
```
Bregenz = N[{FromDMS[{9, 44, 57}], FromDMS[{47, 30, 18}]}];
Konstanz = N[{FromDMS[{9, 11, 00}], FromDMS[{47, 39, 48}]}];
Start = N[{FromDMS[{9, 36, 00}], FromDMS[{47, 30, 00}]}];
Arrival = N[{FromDMS[{9, 21, 00}], FromDMS[{47, 39, 00}]}];

AncreArr = Arrival; (* Ancre de départ: Longitude, Latitude *)
AncreDep = Start; (* Ancre d'arrivée: Longitude, Latitude *)
GraphAttributes = {16, 6, 0.25}; (* # Pas, # Options, Ratio δ Options / δ Pas *)

{Locations, BormesLocations} = GrapheGeneration[AncreDep, AncreArr, GraphAttributes];
{Maxi, Maxj, v} = Dimensions[Locations];

DeparturePoint = {1, 6}; (* #, # *)
ArrivalPoint = {Maxi, 4}; (* #, # *)

{DepartureLocation, ArrivalLocation} =
  {Locations[[ DeparturePoint[[1]], DeparturePoint[[2]] ]], Locations[[ ArrivalPoint[[1]], ArrivalPoint[[2]] ]]};
InDomain[{θ_, ϕ_}] := (LongMin + δlong ≤ θ) && (θ ≤ LongMax - δlong) && (LatMin + δlat ≤ ϕ) && (ϕ ≤ LatMax - δlat);
InTruth = Table[InDomain[Locations[[i, j]]], {i, 1, Maxi}, {j, 1, Maxj}];

{ForecastDay, CosmoStart, CosmoEnd} = TimeTuning ;(* String, GMT, GMT *)
{Spread, DepartureTime} = {3, 12.20}; (* #, CosmoStart < GMT < CosmoEnd *)
```

- **Construction de la matrices des champs de vent sur le graphe**

```
WindinTime = Table[BuildWindTable[Ech], {Ech, 0, 24}];
```

■ Décision: construction de l'arbre des options et choix de la route; présentations graphiques

```
(* Calcul direct *)
{Réseau, Analyse} = ProgrammationDynamique[DeparturePoint[[2]], DepartureTime, Spread];
DimD = First[Dimensions[Analyse]];
(* Calcul rétrograde *)
{Route, Logbuch} = WohinDesWegs;
{Heading, Book} = LogBook;
(* Présentation graphique*)
GrapheFieldDynamique[DepartureTime]
(*Impression du logbook*)
Print[Book];
```



```
{{1, {[ 12h 12mm ], [ 47° 30m 14s ], [ 9° 36m 8s ], [ Kompass: 276° ], [ Speed: 0.928207 Kt ]}},
 {2, {[ 13h 6mm ], [ 47° 30m 20s ], [ 9° 34m 55s ], [ Kompass: 337° ], [ Speed: 1.97055 Kt ]}},
 {3, {[ 13h 40mm ], [ 47° 31m 22s ], [ 9° 34m 16s ], [ Kompass: 326° ], [ Speed: 2.40628 Kt ]}},
 {4, {[ 14h 4mm ], [ 47° 32m 9s ], [ 9° 33m 28s ], [ Kompass: 312° ], [ Speed: 2.42695 Kt ]}},
 {5, {[ 14h 25mm ], [ 47° 32m 43s ], [ 9° 32m 32s ], [ Kompass: 312° ], [ Speed: 2.37224 Kt ]}},
 {6, {[ 14h 46mm ], [ 47° 33m 17s ], [ 9° 31m 36s ], [ Kompass: 312° ], [ Speed: 2.25754 Kt ]}},
 {7, {[ 15h 9mm ], [ 47° 33m 51s ], [ 9° 30m 39s ], [ Kompass: 326° ], [ Speed: 2.1379 Kt ]}},
 {8, {[ 15h 36mm ], [ 47° 34m 38s ], [ 9° 29m 52s ], [ Kompass: 345° ], [ Speed: 1.43988 Kt ]}},
 {9, {[ 16h 30mm ], [ 47° 35m 54s ], [ 9° 29m 21s ], [ Kompass: 294° ], [ Speed: 1.46715 Kt ]}},
 {10, {[ 17h 3mm ], [ 47° 36m 14s ], [ 9° 28m 16s ], [ Kompass: 312° ], [ Speed: 1.52018 Kt ]}},
 {11, {[ 17h 36mm ], [ 47° 36m 48s ], [ 9° 27m 20s ], [ Kompass: 294° ], [ Speed: 1.38088 Kt ]}},
 {12, {[ 18h 11mm ], [ 47° 37m 7s ], [ 9° 26m 15s ], [ Kompass: 326° ], [ Speed: 1.15438 Kt ]}},
 {13, {[ 19h 1mm ], [ 47° 37m 55s ], [ 9° 25m 27s ], [ Kompass: 294° ], [ Speed: 1.4016 Kt ]}},
 {14, {[ 19h 35mm ], [ 47° 38m 15s ], [ 9° 24m 22s ], [ Kompass: 312° ], [ Speed: 1.56249 Kt ]}},
 {15, {[ 20h 7mm ], [ 47° 38m 49s ], [ 9° 23m 26s ], [ Kompass: 326° ], [ Speed: 0.820925 Kt ]}},
 {16, {[ 21h 18mm ], [ 47° 39m 37s ], [ 9° 22m 38s ], [ Kompass: 277° ], [ Speed: 1.28654 Kt ]}}}
```

GrapheFieldStatique[CosmoStart + 22.05]