Projet de fin d'études - Élaboration d'un GPS

VIDAL Antoine
Encadrant : MANOUSSAKIS George
Référent : PILARD Laurence

ISTY Vélizy, 10-12 Avenue de l'Europe, 78140 Vélizy-Villacoublay

IATIC5 2022-2023

Sommaire

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Sommaire

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Introduction — Élaboration du sujet

- Langage choisi : Python en raison de sa popularité. Tkinter a été choisi pour l'interface graphique en raison de la simplicité de l'application réalisée.
- Sujet : choix un algorithme abordé à l'ISTY pour mettre en pratique les connaissances acquises
- Tuteur : professeur ayant enseigné l'algorithme

Introduction — Exemple d'utilisation

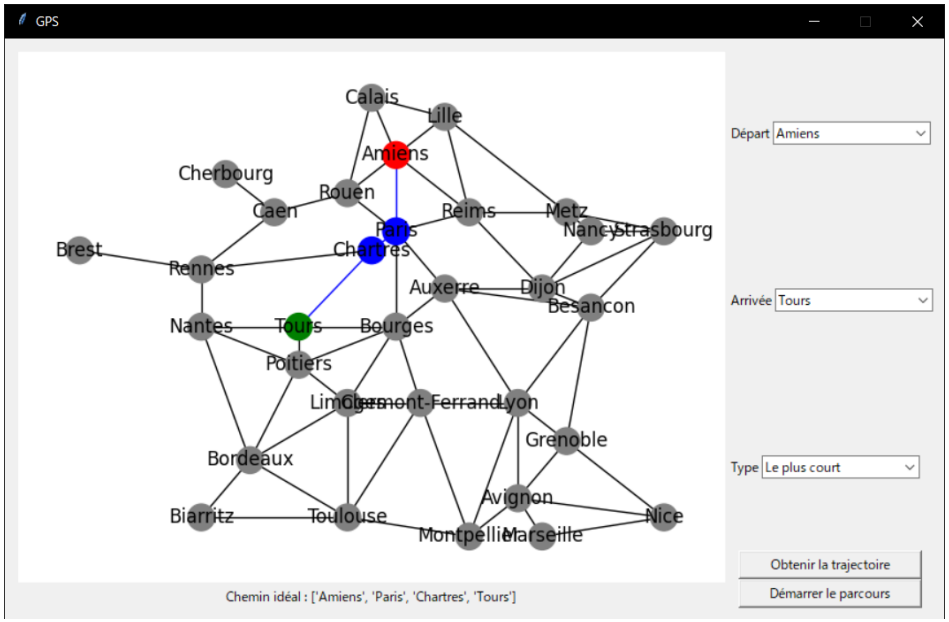

Figure: Choix de l'itinéraire

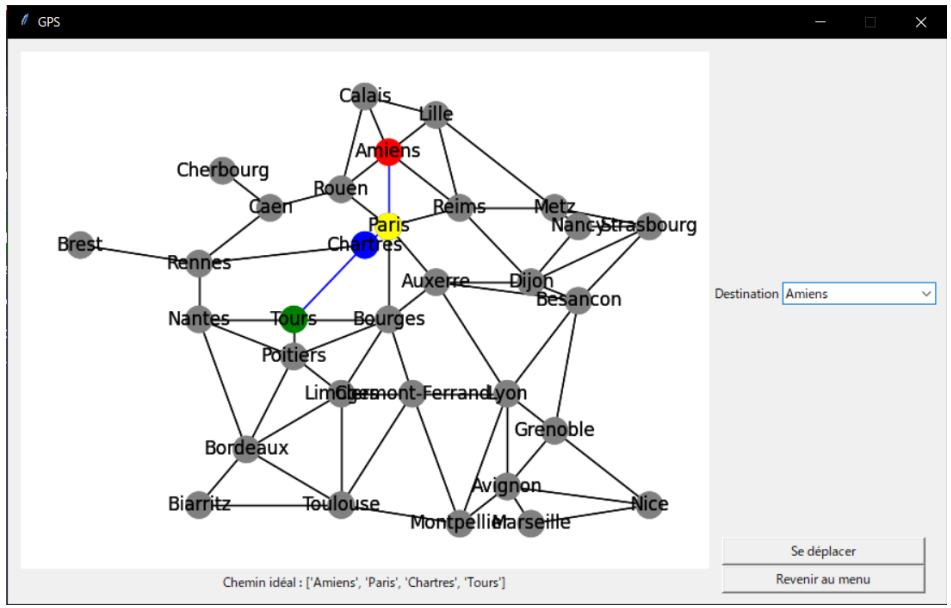

Figure: Choix de la prochaine ville

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Sommaire

UVSQ université PARIS-SACLAY | ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Structure des données hors Python — Carte de la France

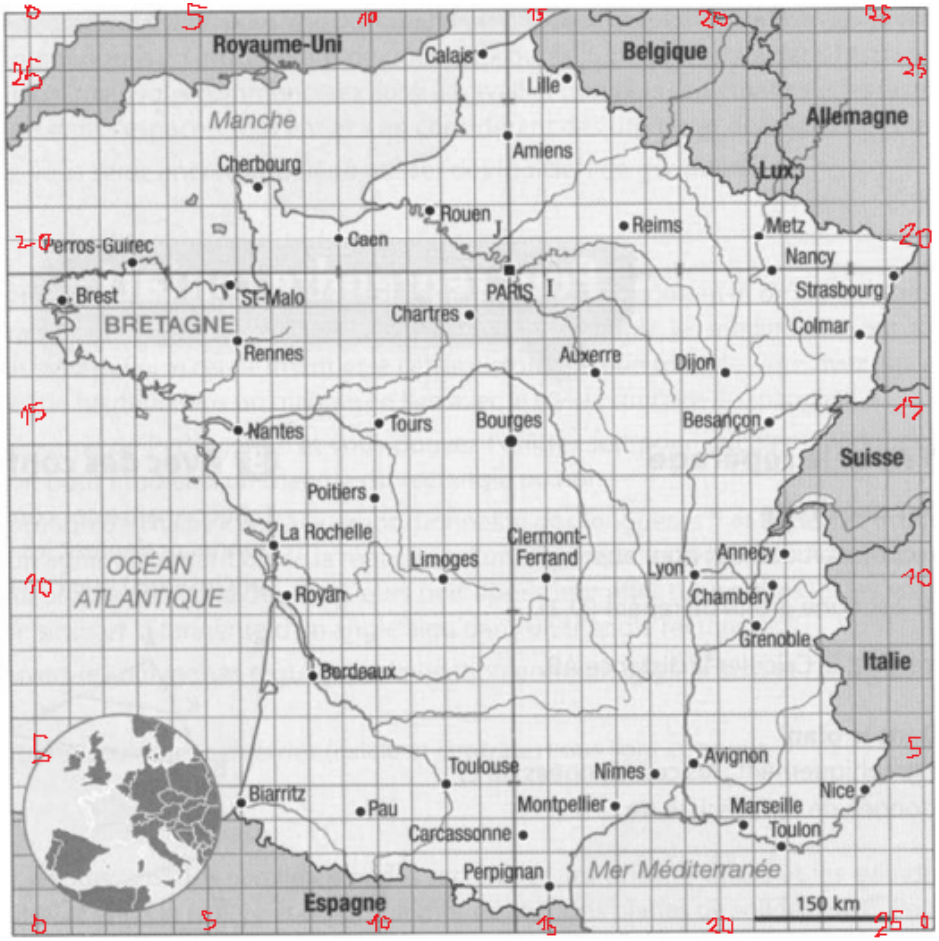

Figure: Carte originale servant de base pour notre application[1]

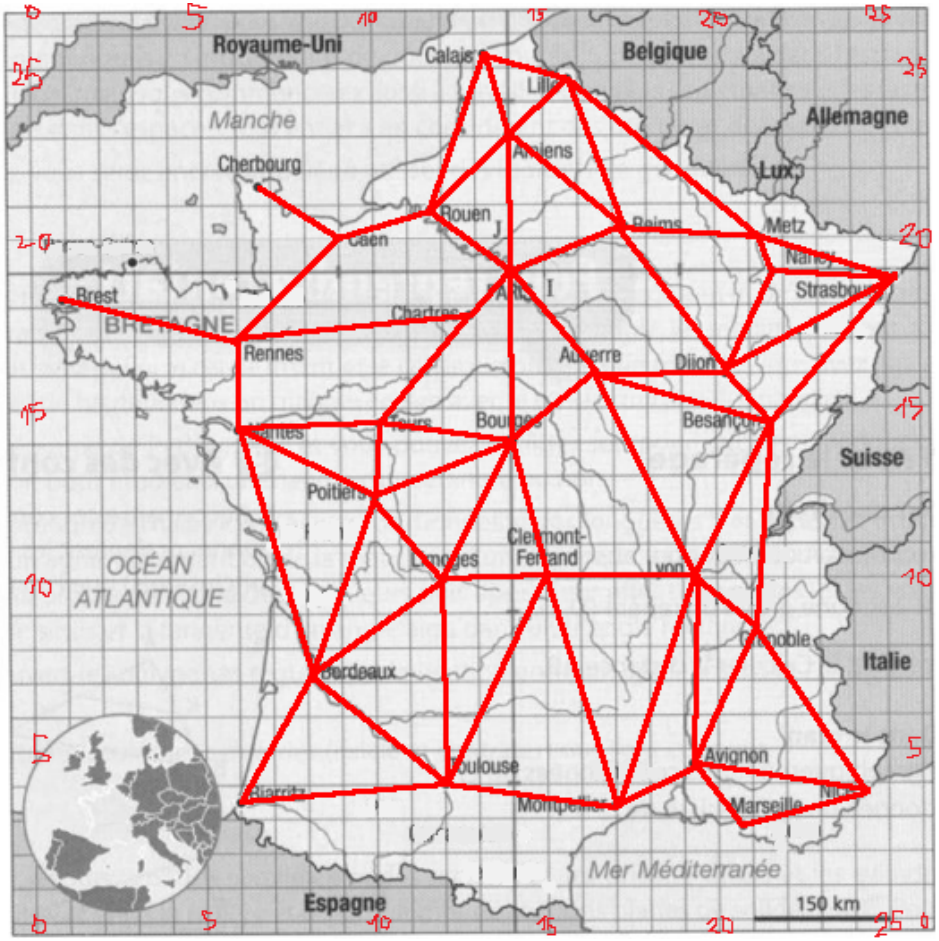

Figure: Carte modifiée utilisée pour notre application

[1] https://www.lewebpedagogique.com/mathasion/files/2013/09/2nde_TP1_

Structure des données hors Python — Format des fichiers

Ville	x (abscisse)	y (ordonnée)
Amiens	14	23
Auxerre	16	16
Avignon	19	5
Besancon	22	15
Biarritz	6	4
Bordeaux	8	7
Bourges	14	14
Brest	1	18
Caen	9	20
Calais	13	26
Chartres	13	18
Cherbourg	7	22

Table: Fichier .csv comportant les positions des villes

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Structure des données hors Python — Format des fichiers

Tableau	Ville 1	Ville 2	Ville 3	Ville 4	Ville 5	Ville 6	Ville 7
Ville 1	0	1	0	0	0	0	2
Ville 2	1	0	0	2	0	0	0
Ville 3	0	0	0	0	0	4	4
Ville 4	0	2	0	0	5	0	0
Ville 5	0	0	0	5	0	0	0
Ville 6	0	0	4	0	0	0	0
Ville 7	2	0	4	0	0	0	0

Table: Fichier .csv comportant les arêtes entre les villes

Sommaire

UVSQ université PARIS-SACLAY | ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Structure des codes Python — Main

Instancier les différents modules Modèle-Vue-Contrôleur et se charger de mettre fin au processus.

```
def quitter():
    if messagebox.askyesno("Quitter", "Voulez-vous quitter?"):
        app.destroy()

app.protocol("WM_DELETE_WINDOW", quitter())
```

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Modèle - Définir et manipuler les données non visibles par l'utilisateur

```
df_towns_edges = pd.read_csv("towns_edges_2d_array.csv")
df_towns_edges = df_towns_edges.set_index("Town")

for rowIndex, row in df_towns_edges.iterrows():
    for columnIndex, value in row.items():
        if value > 0:
            controller.Controller.G.add_edge(rowIndex,
            ↪  columnIndex, weight=value)
```

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Modèle - Déterminer le chemin idéal selon les critères choisis par l'utilisateur

```
def compute_shortest_path(self, start, arrival, type_path):
    controller.Controller.start_town = start
    controller.Controller.arrival_town = arrival
    controller.Controller.shortest_path_type_value = type_path

    if controller.Controller.shortest_path_type_value ==
    ↪  controller.Controller.shortest_path_type_list[0]:
        controller.Controller.shortest_path =
        ↪  nx.dijkstra_path(controller.Controller.G, start, arrival)
    elif controller.Controller.shortest_path_type_value ==
    ↪  controller.Controller.shortest_path_type_list[1]:
        controller.Controller.shortest_path =
        ↪  nx.shortest_path(controller.Controller.G, start, arrival)
    controller.Controller.shortest_path_edges = [(controller.Controller
    ↪  .shortest_path[i],controller.Controller.shortest_path[i+1]) for
    ↪  i in range(len(controller.Controller.shortest_path)-1)]
```

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Vue - Partie droite de l'écran : actions de l'utilisateur

Construction des outils permettant d'interagir avec l'utilisateur

```
cmb_town_start = Combobox(master=self.frm_town_start, state="readonly",
↪  values=controller.Controller.list_towns_text)
cmb_town_start.set(controller.Controller.list_towns_text[0])

btn_confirm = Button(frm_ui_choose_route,text="Obtenir la trajectoire",
command=lambda:controller.Controller.get_shortest_path(controller.Contr
↪  oller, cmb_town_start.get(), cmb_town_arrival.get(),
↪  cmb_shortest_path_type.get()))
```

Figure: Boîte déroulante

Figure: Bouton

Vue - Partie droite de l'écran : actions de l'utilisateur

Réinitialisation des données une fois la destination atteinte

```
if(controller.Controller.current_town ==
↪  controller.Controller.arrival_town):

  messagebox.showinfo("Information", "Vous êtes arrivé.")
  controller.Controller.shortest_path.clear()
  View.lbl_best_path.configure(text="Chemin idéal :
  ↪  {}".format(controller.Controller.shortest_path))
  View.frm_ui_moving.pack_forget()
  View.frm_ui_choose_route.pack(fill=Y, side=RIGHT, expand=True)
```

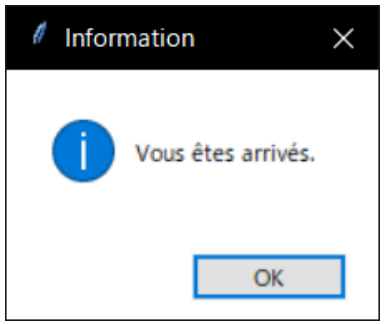

Figure: Message d'information indiquant l'arrivée à destination

Vue - Partie droite de l'écran : actions de l'utilisateur

Cas particulier où l'utilisateur se trompe de chemin

Cas qui semble compliqué à première vue à résoudre

Toutefois cela consiste simplement en un changement d'argument lors du calcul du chemin idéal : la ville de départ devient la ville actuelle

```
if(controller.Controller.current_town not in
↪  controller.Controller.shortest_path):

messagebox.showwarning("Chemin optimal non pris", "Vous avez emprunté
↪  un chemin non optimal. Il est possible que vous vous soyez trompé
↪  de chemin. Nous allons déterminer un nouveau chemin optimal depuis
↪  votre position actuelle.")

controller.Controller.get_shortest_path(controller.Controller.current_t⌋
↪  own, controller.Controller.arrival_town,
↪  controller.Controller.shortest_path_type_value)
```

UVSQ ISTY Institut des Sciences et Techniques des Yvelines université PARIS-SACLAY CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Vue - Partie gauche de l'écran : affichage du graphe et du chemin idéal

Afficher le graphe à chaque modification de celui-ci

```
def update_image_France():
    View.img_France =
    ↪  PIL.ImageTk.PhotoImage(PIL.Image.open("france_graphe.png"))
    View.lbl_France.configure(image=View.img_France)
```

Contrôleur - Obtenir le meilleur chemin

```
def get_shortest_path(self, start, arrival, type_path):
    if(start == arrival):
        messagebox.showwarning("Attention", "Le départ et la
        ↪  destination sont identiques.")

    else:
        Controller.current_town = start
        model.Model.compute_shortest_path(model, start, arrival,
        ↪  type_path)
        Controller.draw_graph_France()
```

Contrôleur - Modifier le graphe

```
def draw_graph_France():
    color_map = ["gray"]*len(Controller.G.nodes())
    for i, node in enumerate(Controller.G.nodes()):
        if node == Controller.start_town:
            color_map[i] = "red"
        elif node == Controller.arrival_town:
            color_map[i] = "green"
        elif node == Controller.current_town:
            color_map[i] = "yellow"
        elif node in Controller.shortest_path:
            color_map[i] = "blue"

    edge_color_list = ["black"]*len(Controller.G.edges())
    for i, edge in enumerate(Controller.G.edges()):
        if edge in Controller.shortest_path_edges or (edge[1],edge[0]) in
        ↪  Controller.shortest_path_edges:
            edge_color_list[i] = "blue"

    view.View.lbl_best_path.configure(text="Chemin idéal :
    ↪  {}".format(Controller.shortest_path))
    nx.draw(Controller.G, Controller.pos, with_labels=True,
    ↪  node_color=color_map, edge_color=edge_color_list)
    plt.savefig("france_graphe.png")
    view.View.update_image_France()
```

UVSQ ISTY
université PARIS-SACLAY
Institut des Sciences et Techniques des Yvelines
CAMPUS DE MANTES EN YVELINES
CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Sommaire

UVSQ université PARIS-SACLAY ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES

Conclusion

- Projet très enrichissant : mise en pratique d'algorithmes appris et familiarisation avec de nouvelles librairies

Conclusion

- Projet très enrichissant : mise en pratique d'algorithmes appris et familiarisation avec de nouvelles librairies
- Satisfait des résultats obtenus : vision initiale du projet similaire au produit final

Conclusion

- Projet très enrichissant : mise en pratique d'algorithmes appris et familiarisation avec de nouvelles librairies
- Satisfait des résultats obtenus : vision initiale du projet similaire au produit final
- Souhait de s'orienter vers des métiers au fonctionnement similaire : développer des applications tout en se familiarisant avec de nouveaux outils

UVSQ université PARIS-SACLAY | ISTY Institut des Sciences et Techniques des Yvelines CAMPUS DE MANTES EN YVELINES CAMPUS DE SAINT-QUENTIN-EN-YVELINES