

Exercice 1. Implémentation de l'algorithme de Berge

Question 1 – Voir les question intermédiaire ci-dessous.

Question 2 –

```
let make_graphe (g_bip: graphe_bip): graphe =
  let g = {
    mat_adj = Array.make_matrix g_bip.n0 g_bip.n1 0;
    s_adj = Array.make g_bip.n0 true;
    t_adj = Array.make g_bip.n1 true;
  } in
  List.iter (fun (s1,s2) -> g.mat_adj.(s1).(s2) <- 1) g_bip.arettes;
  g;;
```

Question 3 –

```
let get_couplage (g: graphe): (int*int) list =
  let n0 = Array.length g.s_adj in
  let n1 = Array.length g.t_adj in
  let coupl = ref [] in
  for u0 = 0 to n0 - 1 do
    for u1 = 0 to n1 - 1 do
      if g.mat_adj.(u0).(u1) = -1 then
        coupl := (u0,u1) :: !coupl
    done;
  done;
  !coupl;;
```

Question 4 –

```
(* c = chaine M-augmentante dont le premier sommet appartient à S0 *)
(* On remarque que les nouveaux sommets saturés sont les extrémités de
la chaine *)
let maj_graphe (g: graphe) (c: int list): unit =
  let rec aux = function
    | u0 :: u1 :: [] ->
      g.mat_adj.(u0).(u1) <- -1;
      g.t_adj.(u1) <- false
    | u0 :: u1 :: v0 :: q ->
      g.mat_adj.(u0).(u1) <- -1;
      g.mat_adj.(v0).(u1) <- 1;
      aux (v0 :: q)
    | _ -> failwith "maj_graphe: liste vide ou de taille impaire en entrée"
  in
  aux c;
  g.s_adj.(List.hd c) <- false;;
```

Question 5.a –

```
let rec list_init (n: int) (pred: int -> bool): int list = match n with
| -1 -> []
| n when pred n -> n :: list_init (n-1) pred
| n -> list_init (n-1) pred;;
```

Question 5.b – On effectue un parcours en profondeur du graphe. Pour cela, on utilise une variable « vis: bool array array » telle que :

- vis.(0).(i) indique si le sommet numéro i de S_0 a déjà été visité.
- vis.(1).(j) indique si le sommet numéro j de S_1 a déjà été visité.

La fonction voisins renvoie la liste des sommets non visités voisins de u.

Le paramètre $k \in \{0, 1\}$ en entrée indique si u appartient à S_0 ou S_1 .

La fonction parc_prof renvoie un chemin de s vers t dans g sous la forme d'une liste chaînée de sommets (ou la liste vide si un tel chemin n'a pas été trouvé).

```
let voisins (g: graphe) (u: int) (vis: bool array array) (k: int) =
  if k = 0 then begin
    let n1 = Array.length vis.(1) in
    let pred v = g.mat_adj.(u).(v) = 1 in
    list_init (n1-1) pred
  end
  else begin
    let n0 = Array.length vis.(0) in
    let pred v = g.mat_adj.(v).(u) = -1 in
    list_init (n0-1) pred
  end;;
```

```
let get_CMA (g: graphe) =
  let n0 = Array.length g.s_adj in
  let n1 = Array.length g.t_adj in
  let vis = [|Array.make n0 false; Array.make n1 false|] in

  let rec parc_prof (li: int list) (k: int): int list = match li with
  | [] -> []
  | u :: q when k = 1 && g.t_adj.(u) -> [u]
  | u :: q when vis.(k).(u) -> parc_prof q k
  | u :: q ->
    vis.(k).(u) <- true;
    let c = parc_prof (voisins g u vis k) (1-k) in
    if c = [] then parc_prof q k else u :: c
  in

  let pred u0 = g.s_adj.(u0) in
  parc_prof (list_init (n0-1) pred) 0;;
```

Question 6 –

```
let berge (g_bip: graphe_bip) =
  let g = make_graphe g_bip in
  let c = ref (get_CMA g) in
  while !c <> [] do
    maj_graphe g !c;
    c := get_CMA g;
  done;
  get_couplage g;;
```

Exercice 2. Étude de l'algorithme de Berge

Question 1.a – Comme γ est M -augmentante, on a :

$$\begin{cases} a_i \notin M & \text{si } i \text{ est impair} \\ a_i \in M & \text{si } i \text{ est pair} \end{cases}$$

De plus, cette chaîne relie deux sommets M -exposés, en particulier $a_\ell \notin M$. La longueur ℓ est donc impaire et peut s'écrire $\ell = 2r + 1$. D'où :

$$M' = (M \setminus \{a_{2j} : j \in \llbracket 1; r \rrbracket\}) \cup \{a_{2j+1} : j \in \llbracket 0; r \rrbracket\}.$$

et :

$$|M'| = |M| - r + (r + 1) = |M| + 1.$$

Question 1.b – Il faut montrer que chaque sommet $u \in S$ apparaît au plus une fois dans les arêtes de M' . On traite plusieurs cas :

- Si u n'appartient pas à la chaîne γ , alors u n'apparaît dans aucun des a_i . Il apparaît autant de fois dans M' que dans M , c'est à dire au plus une fois.
- Si u est le premier sommet de γ . Alors par définition d'une chaîne M -augmentante, u est M -exposé, c'est à dire qu'il n'apparaît pas dans M . De plus, u apparaît dans a_1 et comme γ est une chaîne élémentaire il n'apparaît dans aucun autre des a_j . Finalement, u apparaît exactement une fois dans M' .
- Si u est le dernier sommet de γ , on procède de manière similaire au cas précédent.
- Sinon, u est un sommet de γ mais n'est ni le premier, ni le dernier sommet. Alors la chaîne γ rejoint u par l'une des arêtes a_j et en repart par l'arête a_{j+1} . On a donc $u \in a_j$, $u \in a_{j+1}$ et $u \notin a_{j'}$ pour $j' \notin \{j, j+1\}$. Comme γ est une chaîne M -augmentante, on a $a_j \in M$ et $a_{j+1} \notin M$ ou inversement. Supposons $a_j \in M$ et $a_{j+1} \notin M$ (l'autre cas est symétrique). Le sommet u apparaît dans l'arête $a_j \in M$ mais dans aucune autre arête de M . Ainsi :

$$M' = ((M \setminus \{a_j\}) \cup \{a_{j+1}\}) \Delta \{a_1, \dots, a_{j-1}\} \Delta \{a_{j+2}, \dots, a_\ell\}$$

et u apparaît exactement une fois dans M' .

Question 1.c – Soient n et m le nombre de sommets et d'arêtes dans G . Un couplage M vérifie $|M| \leq m$ et $|M| \leq n/2$. On en déduit que le nombre de tours de boucle est inférieur à $\min(m, n/2)$ donc l'algorithme termine.

Question 2 – Pour chercher une chaîne augmentante, on effectue un parcours, soit un temps d'exécution en $\mathcal{O}(\max(n, m))$. De plus, le nombre de tours de boucle est en $\mathcal{O}(\min(n, m))$. Finalement le temps d'exécution est en $\mathcal{O}(\min(n, m) \cdot \max(n, m)) = \mathcal{O}(nm)$.

Question 3.a – Supposons $a_1 \in M_1$. Puisque les arêtes a_1 et a_2 se suivent dans la chaîne γ , elles ont un sommet en commun noté u . Comme u apparaît au plus une fois dans M_1 et que $u \in a_1$, on en déduit que $a_2 \notin M_1$ et donc $a_2 \in M_2$.

L'autre implication se traite de manière similaire.

Question 3.b – Soit γ la plus grande chaîne élémentaire de $I(G', R)$ et ℓ sa longueur. Alors γ est de la forme $\gamma = (u_0, u_1, \dots, u_\ell)$.

★ Si $\ell = 0$ alors $\deg(u_0) = 0$ (sinon γ ne serait pas de longueur maximale). Comme $I(G', R)$ est connexe, on en déduit que $I(G', R)$ est de type 1.

★ Sinon, on a $\ell > 0$. Puisque M_1 et M_2 sont des couplages, chaque sommet $u \in R$ apparaît au plus une fois dans M_1 et au plus une fois dans M_2 , c'est à dire que $\deg(u) \leq 2$. Grâce aux arêtes présentes dans γ , on a :

$$1 \leq \deg(u_0) \leq 2, \quad 1 \leq \deg(u_\ell) \leq 2, \quad \deg(u_i) = 2 \text{ pour tout } i \in \llbracket 1, \ell - 1 \rrbracket.$$

★★ Si $\deg(u_0) = \deg(u_\ell) = 1$ alors toutes les arêtes contenant l'un des u_i sont présentes dans γ . Comme $I(G', R)$ est connexe, on a $R = \{u_0, \dots, u_\ell\}$ et donc $I(G', R)$ est de type 2.

★★ Sinon, supposons $\deg(u_0) = 2$ (le cas $\deg(u_\ell) = 2$ se traite de la même façon). Les voisins de u_0 dans $I(G', R)$ sont u_1 et un autre sommet noté $v \neq u_1$. Puisque γ est une chaîne élémentaire de longueur maximum, le sommet v appartient déjà à γ . On a :

- $v \neq u_0$ (car le graphe est biparti, donc ne contient pas de boucle).
- $v \neq u_1$ (voir l'hypothèse ci-dessus).
- $v \notin \{u_2, \dots, u_{\ell-1}\}$ (sinon on aurait $\deg(v) \geq 3$).

En conclusion, $v = u_\ell$ et donc $\deg(u_\ell) = 2$. Puisque $I(G', R)$ est connexe, on a $R = \{u_0, \dots, u_\ell\}$ et son ensemble d'arêtes est composé de l'arête $\{u_0, u_\ell\}$ et des arêtes de γ . Finalement $I(G', R)$ est de type 3.

Question 4.a – Pour commencer, remarquons qu'une arête de $M_1 \Delta M_2$ ne peut pas appartenir à la fois à M_1 et à M_2 . On a :

$$(M_1 \Delta M_2) \cap M_1 = M_1 \setminus (M_1 \cap M_2) \quad (M_1 \Delta M_2) \cap M_2 = M_2 \setminus (M_1 \cap M_2)$$

Donc :

$$|(M_1 \Delta M_2) \cap M_1| = |M_1| - |M_1 \cap M_2| < |M_2| - |M_1 \cap M_2| = |(M_1 \Delta M_2) \cap M_2|$$

Question 4.b – Soient R_1, R_2, \dots les composantes connexes de G' . Dans la suite, on note $G_j = I(G', R_j)$ et A_j l'ensemble des arêtes de G_j . D'après la question 3.b, chaque G_j est de type 1, 2 ou 3 :

- Si G_j est de type 1, alors $A_j = \emptyset$ donc $A_j \cap M_1 = A_j \cap M_2 = \emptyset$.
- Si G_j est de type 3, on note γ le cycle formé par les sommets et les arêtes de G_j . Alors γ est de longueur paire car G_j est biparti. Grâce à la question 3.a, on sait qu'il contient autant d'arêtes de M_1 que d'arêtes de M_2 .

Lorsqu'on ne considère que les sous-graphes G_j de type 1 ou 3, le nombre d'arêtes issues de M_1 est égal au nombre d'arêtes issues de M_2 . Grâce à la question 4.a, on sait donc qu'il existe au moins un G_j de type 2 contenant plus d'arêtes issues de M_2 que d'arêtes issues de M_1 .

Question 4.c – Soit R la composante connexe définie dans la question 4.b et $\gamma = (u_0, u_1, \dots, u_\ell)$ la chaîne de G' composée des sommets et arêtes de $I(G', R)$. Vérifions que γ est bien M_1 -augmentante.

Par la question 3.a, γ alterne entre des arêtes de M_1 et de M_2 . Comme γ contient plus d'arêtes de M_2 , sa longueur ℓ est impaire, elle commence par une arête de M_2 et termine par une arête de M_2 . On a alors :

- $\ell > 0$ par hypothèse.
- u_0 et u_ℓ sont M_1 -exposés. En effet, toutes les arêtes incidentes à u_0 et u_ℓ sont dans $I(G', R)$. La seule arête incidente à u_0 (resp. u_ℓ) est donc la première (resp. dernière) arête de γ . Cette arête n'appartient pas à M_1 .
- Si i est pair alors l'arête $\{u_i, u_{i+1}\} \notin M_1$.
- Si i est impair alors l'arête $\{u_i, u_{i+1}\} \in M_1$.
- γ est bien une chaîne élémentaire.

En conclusion, γ est bien une chaîne M_1 -augmentante.

Question 5 – On montre les deux sens de l'équivalence.

(\Rightarrow) Par contraposée, supposons que G admet une chaîne M -augmentante notée γ et montrons que M n'est pas un couplage maximal de G . D'après les questions 1.a et 1.b, si on note A l'ensemble des arêtes de γ alors l'ensemble $M' = M \Delta A$ est un couplage de G et $|M'| = |M| + 1$. Donc M n'est pas un couplage maximal de G .

(\Leftarrow) Par contraposée, on suppose que M n'est pas un couplage maximal et on montre que G contient une chaîne M -augmentante. Si M n'est pas maximale, c'est qu'il existe un couplage M' vérifiant $|M| < |M'|$. Par la question 4.c, il existe une chaîne M -augmentante dans G .

Question 6 – On utilise l'invariant de boucle :

(\mathcal{P}_i) : L'ensemble M_i est un couplage de G .

On a (\mathcal{P}_0) car $M_0 = \emptyset$ et (\mathcal{P}_i) \Rightarrow (\mathcal{P}_{i+1}) par la question 1.b. Par itération finie, l'ensemble M_k est bien un couplage de G . De plus, comme l'algorithme renvoie M_k , on sait qu'il n'existe pas de chaîne M_k -augmentante dans G . Par la question 5, M_k est bien un couplage maximal dans G .