

Rapport de Projet - Partie 1: Gestion de Projet PERT/GANTT

Nom Prénom

15 avril 2025

1 Introduction

Ce rapport présente l'implémentation d'un outil de gestion de projet utilisant les méthodes PERT (Program Evaluation and Review Technique) et GANTT. Ces techniques permettent de planifier, coordonner et contrôler des projets complexes en décomposant le travail en tâches individuelles, en identifiant leurs relations de dépendance, et en déterminant les délais critiques pour l'achèvement du projet.

L'outil développé dans ce projet offre les fonctionnalités suivantes :

- Création et manipulation de projets avec leurs tâches associées ;
- Calcul des dates au plus tôt et au plus tard pour chaque tâche ;
- Détermination des marges et identification du chemin critique ;
- Génération de visualisations graphiques (graphe PERT et diagramme de Gantt) ;
- Exportation des résultats sous différents formats.

2 Structure du code

Le programme est structuré autour d'une classe principale `ProjetPERT` qui encapsule toute la logique de l'algorithme PERT. Voici les principales méthodes :

Méthodes de base

- `ajouter_tache()`
- `charger_taches_depuis_csv()`
- `charger_taches_depuis_dataframe()`

Méthodes d'analyse

- `construire_graphe()`
- `calculer_dates_tot(), calculer_dates_tard(), calculer_marges()`
- `trouver_chemin_critique()`
- `analyser_projet()`

Méthodes de visualisation

- `afficher_resultats()`
- `dessiner_graphe()`
- `generer_gantt()`
- `exporter_tableau_resultats(), exporter_visualisations()`

3 Exemple 1 : Projet de développement logiciel

Données d'entrée

Tâche	Durée	Antériorités
A (Analyse des besoins)	4	-
B (Spécification technique)	8	-
C (Conception de l'interface)	1	-
D (Validation de l'interface)	1	C
E (Architecture logicielle)	6	A
F (Planification des tests)	3	A
G (Conception BDD)	5	B
H (Développement back-end)	3	E, F, G
I (Développement interface)	1	D
J (Intégration interface)	2	I
K (Tests unitaires)	2	H
L (Tests int. & déploiement)	5	J, K

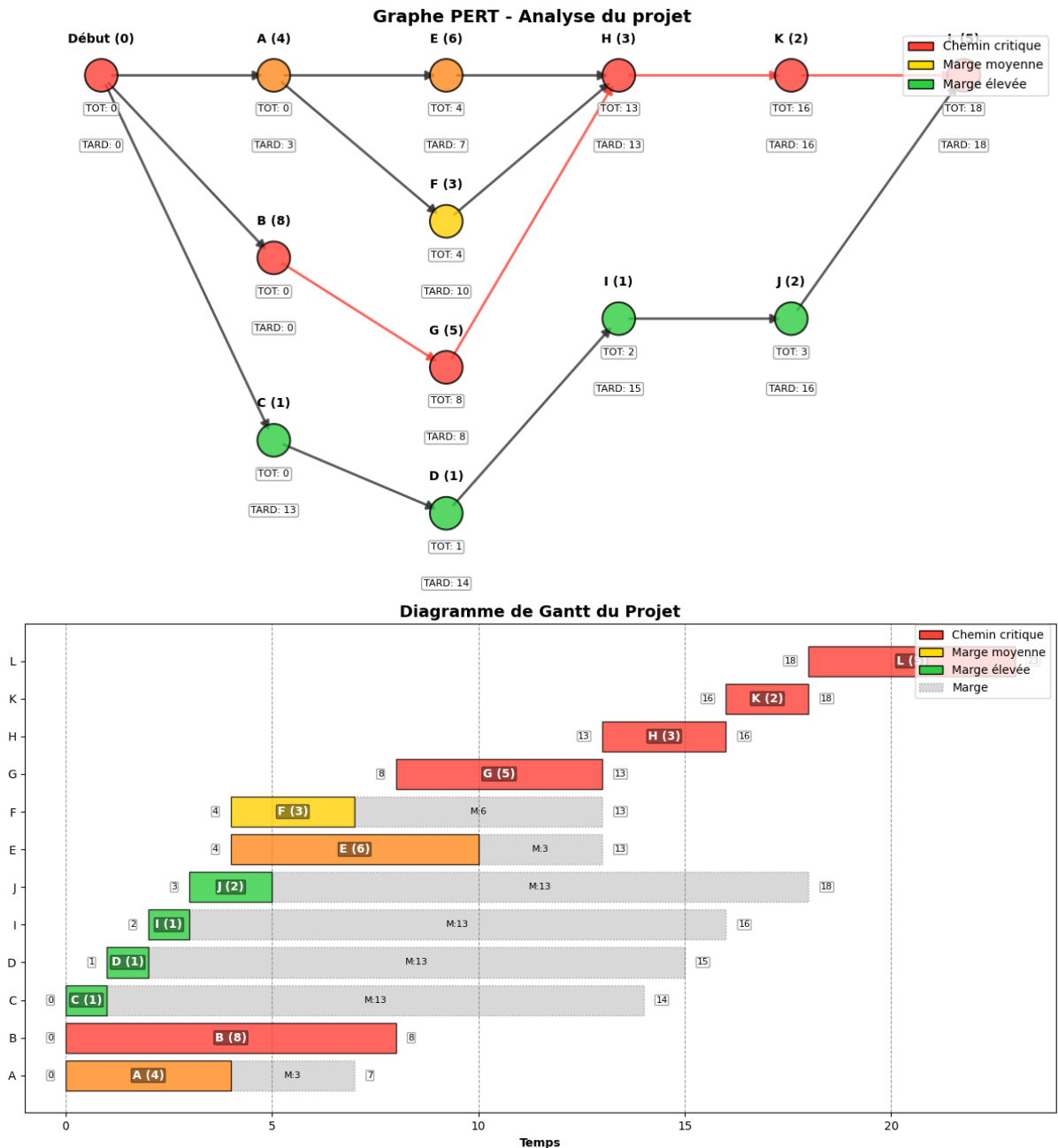
Analyse des résultats

Tâche	Durée	Tôt	Tard	Marge	Critique
A	4	0	0	0	Oui
B	8	0	1	1	Non
C	1	0	10	10	Non
D	1	1	11	10	Non
E	6	4	4	0	Oui
F	3	4	7	3	Non
G	5	8	9	1	Non
H	3	14	14	0	Oui
I	1	2	12	10	Non
J	2	3	15	12	Non
K	2	17	17	0	Oui
L	5	19	19	0	Oui

Chemin critique : $B \rightarrow E \rightarrow H \rightarrow K \rightarrow L$

Durée totale : 24 unités de temps

Visualisations :



Analyse

Les tâches critiques sont principalement centrées sur l'architecture, le développement back-end et les tests. Les étapes d'interface disposent de marges importantes, ce qui permet leur exécution flexible.

4 Exemple 2 : Construction d'une maison

Données d'entrée

Tâche	Durée	Antériorités
A (Préparation)	3	-
B (Fondations)	7	A
C (Murs)	10	B

D (Toiture)	8	C
E (Électricité)	5	C
F (Plomberie)	6	C
G (Isolation)	4	D
H (Cloisons)	7	E, F
I (Menuiseries)	3	D
J (Chauffage)	4	F, G
K (Sol)	5	H, J
L (Peinture)	8	H, I
M (Finitions ext.)	6	I
N (Nettoyage)	2	K, L, M

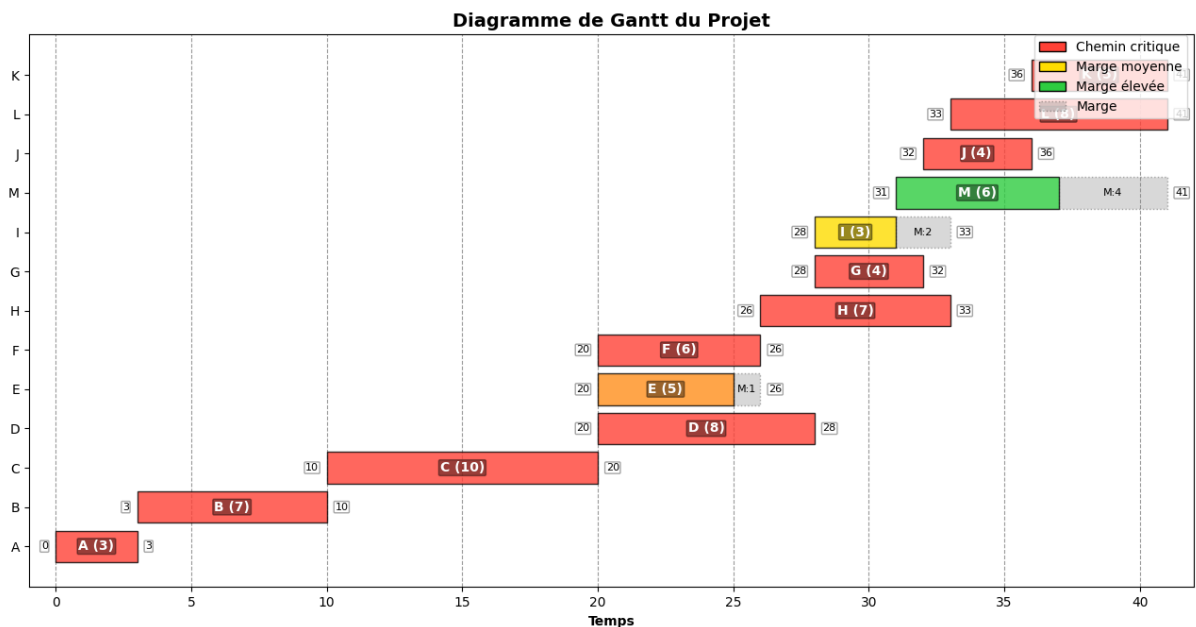
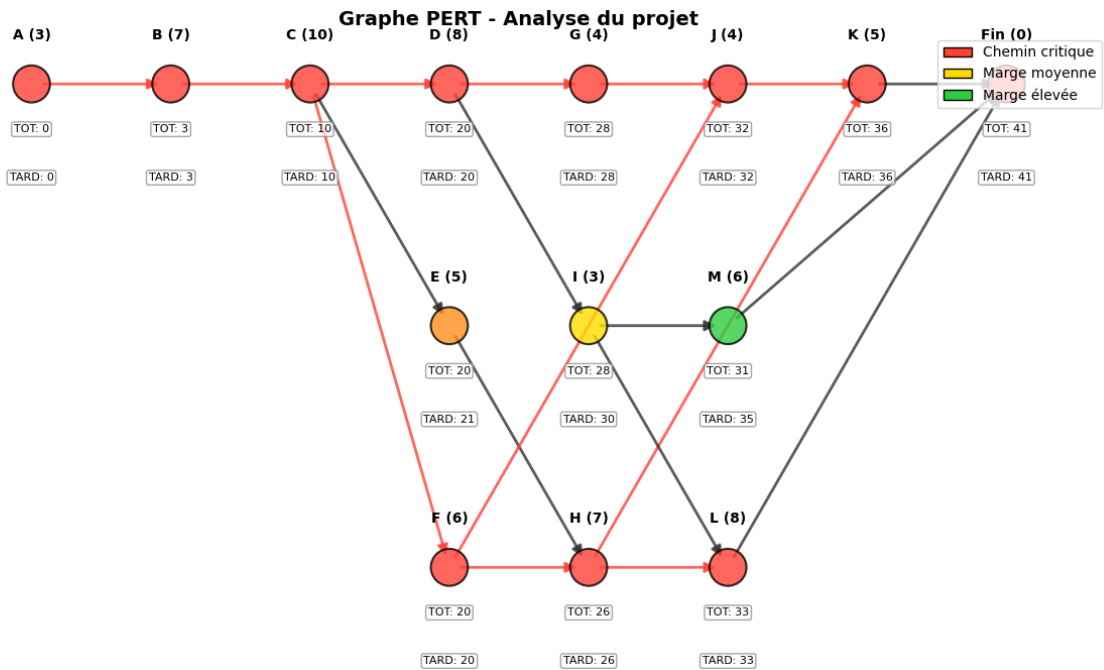
Analyse des résultats

Tâche	Durée	Tôt	Tard	Marge	Critique
A	3	0	0	0	Oui
B	7	3	3	0	Oui
C	10	10	10	0	Oui
D	8	20	20	0	Oui
E	5	20	21	1	Non
F	6	20	20	0	Oui
G	4	28	28	0	Oui
H	7	26	26	0	Oui
I	3	28	34	6	Non
J	4	32	32	0	Oui
K	5	36	36	0	Oui
L	8	33	33	0	Oui
M	6	31	37	6	Non
N	2	41	41	0	Oui

Chemin critique : $A \rightarrow B \rightarrow C \rightarrow D \rightarrow G \rightarrow J \rightarrow K \rightarrow L$

Durée totale : 43 unités de temps

Visualisations :



Analyse

Ce projet suit une séquence fortement linéaire typique du BTP. Les tâches critiques se succèdent de manière rigide, tandis que certaines tâches secondaires comme les menuiseries ou finitions disposent de marges.

5 Conclusion et perspectives

L'implémentation réalisée permet :

- L'analyse de projets PERT de manière automatisée ;
- L'identification des chemins critiques ;
- La génération de plannings Gantt ;
- L'export des résultats et visualisations.

Perspectives d'amélioration

- Intégration de la méthode de la chaîne critique (CCPM)
- Interface utilisateur graphique
- Gestion des ressources et contraintes multiples
- Gestion des incertitudes via simulation Monte-Carlo

Ce projet constitue une base solide pour la planification et l'optimisation temporelle des projets complexes.