

# Projet M2D – 2025-2026

## Modélisation d'un modèle simple de maintenance

Sujet : **Conception d'un simulateur d'endommagement d'une flotte de composants.**  
**Évaluation des politiques de maintenance.**

### Sommaire

On pose le problème .....	1
Premiers attendus.....	1
Inversion du problème (bonus) .....	2
Proposition et formalisation .....	2
Première partie : spécifications techniques.....	2
Seconde partie : développement.....	2
Troisième partie : résultats .....	2

### On pose le problème

Nous ferons l'hypothèse que nous disposons d'une flotte de systèmes avec un composant susceptible de s'endommager puis de tomber en panne. Comme le système est fabriqué par une usine, tous les systèmes et tous les composants n'ont pas le même âge à un instant donné. Mais on considèrera que quand un système a été mis en opération son composant était neuf.

Pour simplifier, dans un premier temps, tous les composants seront de même type en ne posséderont qu'un unique mode de panne. Cette panne peut subvenir aléatoirement à la suite d'un endommagement progressif dû à l'usage du composant. Ces composants ne sont pas utilisés tous à la même fréquence, ni régulièrement. La vitesse de dégradation du composant peut augmenter quand le système qui l'intègre vieillit.

Par ailleurs des inspections quasi régulières sont effectuées par une équipe de maintenance. Lors d'une inspection l'opérateur utilise un logiciel de détection d'usure qui n'est pas parfait, s'il détecte une anomalie, on change ou on répare le composant et le système repart avec un composant neuf.

### Premiers attendus

On s'intéresse au risque de pannes à un instant donné. C'est-à-dire que l'on cherche à chaque instant quelle est le nombre de composants qui vont tomber en panne dans un délai donné. Ce sont des composants qui tombent en panne et donc qui n'ont pas été ni détectés ni réparés. Plus exactement on étudiera la loi de ce nombre de pannes potentielles ce qui nous permettra de construire des intervalles de confiance.

On souhaite ensuite optimiser le risque de panne en jouant sur l'intervalle inter inspection.

On souhaite aussi maîtriser l'effet de la qualité du détecteur en conjonction avec la politique de maintenance appliquée.

Que se passe-t-il si on définit un coût associé à l'inspection, à une réparation et le surcout dû une panne. À partir de quand faut-il toujours inspecter les composants plutôt que d'attendre qu'ils tombent en panne ?

### Inversion du problème

On dispose cette fois d'observations qui sont des dates de réparations, et de pannes ainsi que les vrais usages des systèmes. Le but est cette fois de retrouver les modes de pannes statistiques du composant, pour voir si on peut calibrer le simulateur pour construire des prédictions de risque.

## Proposition et formalisation

Ce projet se décompose en plusieurs étapes

1. Conception d'un simulateur.
2. Génération de données à partir du simulateur.
3. Analyse descriptive des données (comme si elles venaient de systèmes réels).
4. Mise en place d'une mesure de risque, de coût...
5. Étude des effets des différents paramètres sur les indicateurs précédents.
6. Inversion du modèle (retrouver les paramètres d'usure et connaissant les données, la politique de maintenance et l'efficacité du détecteur.

### Première partie : spécifications techniques

Pour la réalisation de cet objectif il faudra faire d'abord une proposition technique sous la forme d'un document qui explique votre démarche, l'architecture du code et des données que vous envisagez, les interfaces graphiques proposées. Vous accompagnerez ce document d'un plan de travail décomposé en lots et d'un diagramme de Gantt pour expliquer la répartition de la charge et les dates de livrables.

### Seconde partie : développement

Dans un second temps vous commencerez à réaliser un code. Vous penserez à produire des exemples applicatifs, des tests permettant de valider votre développement.

### Troisième partie : résultats

Vous réaliserez une présentation PowerPoint de vos travaux pour la restitution. Les codes devront être accessibles et organisés sur une plateforme de développement communautaire (par exemple GitHub).