ÉCOLE NATIONALE SUPERIEURE POLYTECHNIQUE
DE YAOUNDÉ

DEPARTEMENT DE GENIE INFORMATIQUE

INTRODUCTION AUX TECHNIQUES
D'INVESTIGATION NUMERIQUE

EXERCICES D'INVESTIGATION NUMERIQUE

MAKEU TENKU STELY BELVA
CIN-4

Superviseur : M. THIERRY
MINKA

2025-2026

EXERCICES D'INVESTIGATION NUMERIQUE

Partie 1 : Fondements Philosophiques et Épistémologiques

Analyse Critique du Paradoxe de la Transparence

Le terme transparence possède plusieurs significations selon le contexte. Selon le dictionnaire Robert, il s'agit d'un phénomène par lequel les rayons lumineux sont perçus à travers certaines substances. Cette définition physique illustre l'idée de clarté et de visibilité, concept que l'on transpose aisément dans les domaines sociaux et numériques. Dans le contexte de la protection des données, la transparence est un principe selon lequel le traitement des informations personnelles ne doit pas être opaque ou caché : la personne concernée doit être informée de manière claire, accessible et compréhensible. Cependant, le philosophe Byung-Chul Han met en évidence un paradoxe : plus la transparence est érigée en norme absolue, plus elle fragilise la confiance, la liberté et l'intimité qu'elle prétend protéger. Ce paradoxe nous invite à interroger les limites de la transparence et à réfléchir à la manière dont elle peut devenir contre-productive.

Dans la société moderne, la transparence est souvent perçue comme une valeur positive et indispensable. Dans le domaine juridique et numérique, elle garantit le droit à l'information et renforce la confiance entre l'individu et les institutions. Par exemple, le Règlement Général sur la Protection des Données (RGPD) impose aux responsables de traitement d'informer les utilisateurs sur les données collectées, les finalités de leur usage, les personnes avec qui elles sont partagées et la durée de conservation. Cette exigence permet à chacun d'exercer un consentement éclairé et protège la vie privée. Dans les relations sociales et professionnelles, la transparence est associée à la sincérité et à l'authenticité. Montrer ses intentions, expliquer ses choix et rendre compte de ses actions renforce la confiance et facilite la communication. Ainsi, la transparence est présentée comme un outil de protection et de responsabilisation, tant pour l'individu que pour la société.

Pour Byung-Chul Han, la transparence possède un double tranchant.
Lorsqu'elle devient totale et obligatoire, elle se transforme en un instrument de contrôle et d'aliénation. Dans une société où tout doit être visible et vérifiable, la confiance disparaît paradoxalement : si chacun doit prouver sans cesse sa sincérité, personne ne peut plus faire confiance spontanément. La transparence absolue engendre ainsi la méfiance généralisée et réduit la liberté des individus. De plus, l'exposition permanente de l'information réduit l'intimité et la créativité. Dans le domaine numérique, par exemple, les utilisateurs sont incités à partager chaque geste ou opinion, souvent sous l'effet des réseaux sociaux. Cette visibilité constante transforme les individus en surfaces lisses, facilement quantifiables et surveillables, ce qui restreint leur capacité à penser ou agir librement. Le paradoxe est donc que la transparence, censée protéger la personne et renforcer la confiance, produit en réalité une surveillance et un contrôle plus étroits, au détriment de la liberté et de la profondeur des relations humaines. Le paradoxe de la transparence, tel qu'identifié par Byung-Chul Han, révèle les limites d'un idéal poussé à l'extrême. Si la transparence est nécessaire pour protéger les droits, garantir la sécurité et instaurer la confiance, son excès peut générer méfiance, exposition forcée et perte d'intimité. Elle ne consiste donc pas à tout rendre visible, mais à trouver un équilibre entre clarté et opacité, entre information nécessaire et protection de la subjectivité. Dans un monde numérique et hyperconnecté, le véritable défi consiste à concevoir des mécanismes de transparence qui protègent tout

en préservant la liberté, la créativité et la dignité des individus.

Exemple concret

Rendre publiques les dépenses du gouvernement permet aux citoyens de contrôler l'utilisation des fonds publics, favorisant ainsi la responsabilité et la transparence. Cependant, la publication excessive de certaines informations personnelles liées à des fonctionnaires ou des bénéficiaires peut porter atteinte à la vie privée et générer un climat de surveillance et de méfiance. Il est donc crucial de trouver un équilibre entre la transparence des actes gouvernementaux et la protection des données personnelles afin de maintenir la confiance publique.

Solution inspirée de Kant

Pour résoudre ce paradoxe, on peut appliquer l'éthique kantienne : chaque action de transparence doit être guidée par un principe universalisable qui respecte la dignité humaine. Concrètement, cela signifie rendre publics uniquement les éléments qui servent l'intérêt général, tout en protégeant strictement les données personnelles. Cette approche permet de concilier transparence et respect des libertés individuelles, garantissant que la responsabilité publique ne sacrifie pas l'autonomie ou la dignité des citoyens.

Exercice 2 : Transformation Ontologique du Numérique

Question 1 : Heidegger et le numérique

Selon Martin Heidegger, l'être humain se définit par son « être-au-monde », une existence intrinsèquement liée à son environnement, à ses interactions et à sa temporalité. L'individu n'existe pas isolément mais toujours en relation avec le monde qui l'entoure, et sa présence se comprend à travers ses engagements et ses expériences. À l'ère numérique, cette conception subit une transformation majeure : l'existence humaine se manifeste désormais aussi par des traces immatérielles, laissées dans les environnements numériques. Ces empreintes digitales numériques, qu'il s'agisse de messages, de publications ou de comportements enregistrés, constituent une extension de l'« être-au-monde », rendant l'individu à la fois présent et documenté dans des espaces virtuels.

Question 2 : Exemple de profil social

Un profil social complet sur une plateforme numérique illustre concrètement cette mutation ontologique. Il agrège des données personnelles, des interactions, des comportements et des préférences, créant une représentation partielle mais détaillée de l'individu. Ce profil incarne ce que l'on peut qualifier d'« être-par-la-trace » : l'existence de l'utilisateur se trouve partiellement projetée dans le monde numérique, de manière durable et souvent indépendante de sa volonté consciente. Chaque action, chaque partage et chaque comportement enregistré contribuent à construire une identité numérique qui devient une extension de la présence physique et sociale, transformant la manière dont l'individu est perçu et évalué.

Question 3 : Impact sur la preuve légale

Cette évolution ontologique transforme profondément la notion de preuve dans le domaine juridique. Alors que la preuve traditionnelle reposait sur des objets tangibles ou des témoignages directs, les environnements numériques offrent des traces immatérielles riches d'information mais également susceptibles de manipulation ou d'interprétation erronée. La preuve numérique exige donc de nouvelles méthodes de collecte, de vérification et de protection pour garantir sa fiabilité et sa légitimité. Cette réévaluation implique de concilier l'exploitation de ces données avec le respect des droits fondamentaux, notamment la vie privée, et de repenser les cadres légaux afin que l'« être-par-la-trace » serve à renforcer la justice plutôt qu'à la compromettre.

Partie 2 : Mathématiques de l'Investigation

Exercice 3 : Analyse de l'entropie

Question 1 : Téléchargement des fichiers

Pour cette analyse, nous utilisons trois types de fichiers afin d'étudier l'entropie de données numériques :

— Un document texte simple (`document.txt`) contenant du texte clair.
— Une image JPEG (`photo.jpg`) représentant des données compressées.
— Un fichier chiffré AES (`document.txt.enc`) obtenu en chiffrant le document texte.

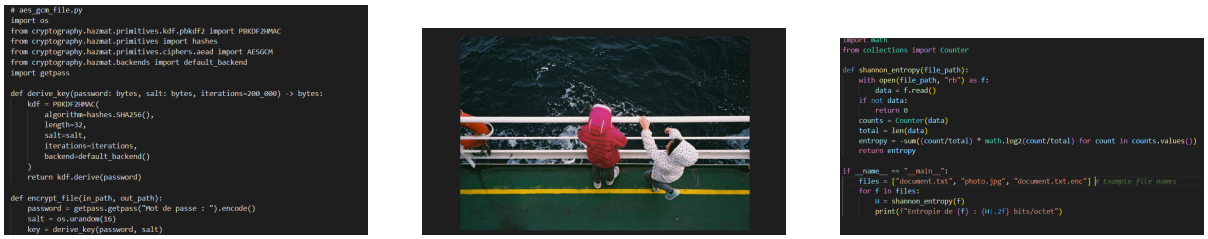


FIGURE 1 – Exemples des fichiers utilisés : texte, image JPEG et fichier chiffré AES

Ces fichiers serviront de base pour le calcul de l'entropie de Shannon et pour illustrer la différence de caractère aléatoire entre un texte simple, une image compressée et un fichier chiffré.

Question 2 : Script Python

```
import math
from collections import Counter

def shannon_entropy(file_path):
with open(file_path, "rb") as f:
data = f.read()
if not data:
return 0
counts = Counter(data)
```

```
                    total = len(data)
                    entropy = -sum((count/total) * math.log2(count/
                        total) for count in counts.values())
                    return entropy

                    if __name__ == "__main__":
                    files = ["document.txt", "photo.jpg", "document.txt
                        .enc"]
                    for f in files:
                    H = shannon_entropy(f)
                    print(f"Entropie de {f} : {H:.2f} bits/octet")
```

Question 3 : Résultats

Pour analyser la nature des fichiers dans le cadre de l'investigation numérique, nous avons calculé l'entropie de trois types de fichiers : un document texte (document.txt), une image JPEG (photo.jpg) et un fichier chiffré AES (document.txt.enc). Les résultats obtenus sont respectivement de 4,88 bits/octet pour le texte, 7,98 bits/octet pour l'image et 7,91 bits/octet pour le fichier chiffré. L'entropie du document texte, plus faible, reflète la présence de motifs répétitifs et une structure prévisible typique des fichiers non compressés. À l'inverse, l'entropie élevée de l'image JPEG traduit l'effet de la compression, qui rend les données plus aléatoires. Enfin, l'entropie très proche de 8 bits/octet du fichier chiffré montre que le chiffrement AES a produit un flux quasi-aléatoire, rendant le contenu indiscernable et garantissant la confidentialité. Ces observations confirment que l'entropie peut être utilisée comme un indicateur fiable pour détecter automatiquement des fichiers chiffrés ou compressés dans le cadre d'analyses forensiques.

Question 4 : Seuil de chiffrement

Un fichier avec une entropie ≥ 7.8 bits/octet est probablement chiffré, car proche de l'entropie maximale (8 bits/octet).

Exercice 4 : Théorie des graphes en investigation criminelle

Pour analyser les communications entre individus, nous construisons un graphe où chaque nœud représente une personne et chaque arête représente une communication téléphonique. Les poids des arêtes correspondent au nombre d'appels échangés. Cette représentation permet d'étudier les relations et de détecter les acteurs clés du réseau.

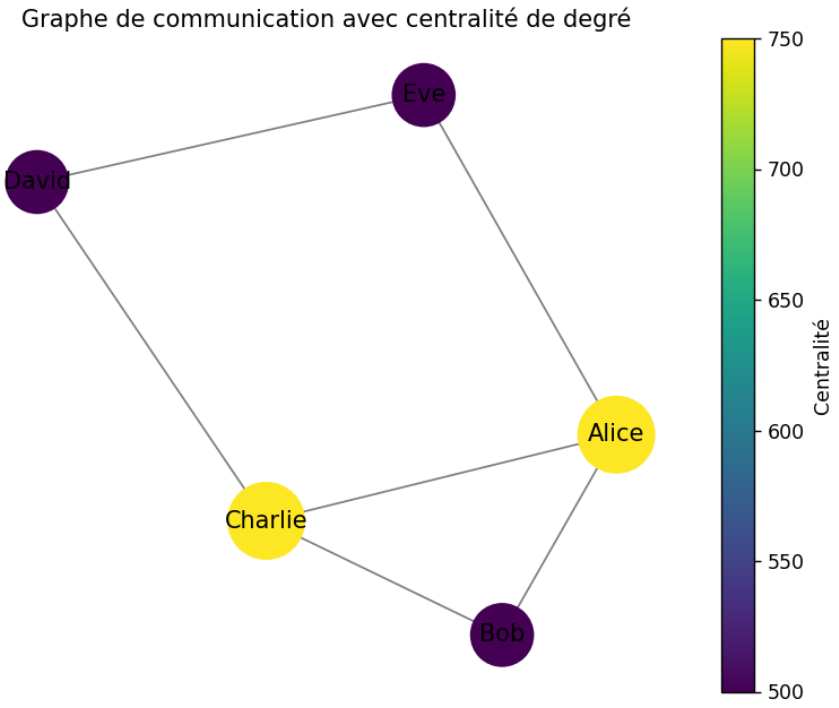
Nous calculons plusieurs métriques de centralité :

- **Centralité de degré** : importance d'un nœud selon le nombre de connexions directes.
- **Centralité d'intermédiarité** (betweenness) : contrôle potentiel du flux d'information à travers le nœud.
- **Centralité de proximité** (closeness) : rapidité avec laquelle un nœud peut atteindre tous les autres nœuds.

Les nœuds ayant la centralité d'intermédiarité la plus élevée sont considérés comme critiques (algorithme de Freeman).

Listing 1 – Construction et analyse du graphe

```
import networkx as nx
import matplotlib.pyplot as plt

#Exemple de donnees (source, cible, poids)
data = [
("Alice", "Bob", 3),
("Alice", "Charlie", 2),
("Bob", "Charlie", 5),
("Charlie", "David", 1),
("David", "Eve", 2),
("Eve", "Alice", 1)
]

#Creation du graphe
G = nx.Graph()
for src, tgt, weight in data:
G.add_edge(src, tgt, weight=weight)

#Calcul des centralites
degree_centrality = nx.degree_centrality(G)
betweenness_centrality = nx.betweenness_centrality(
    G, weight='weight')
closeness_centrality = nx.closeness_centrality(G)

#Identifier les noeuds critiques
critical_nodes = sorted(betweenness_centrality.
    items(), key=lambda x: x[1], reverse=True)
print("Noeuds critiques (intermediarite) :",
    critical_nodes[:3])

# Visualisation du graphe
node_color = [degree_centrality[node]*1000 for node
     in G.nodes()]
node_size = [degree_centrality[node]*2000 for node
    in G.nodes()]

plt.figure(figsize=(8,6))
pos = nx.spring_layout(G, seed=42)
nx.draw_networkx_edges(G, pos, alpha=0.5)
nodes = nx.draw_networkx_nodes(G, pos, node_color=
    node_color,
node_size=node_size, cmap=plt.cm.viridis)
nx.draw_networkx_labels(G, pos)
plt.colorbar(nodes, label="Centralite")
plt.title("Graphe de communication avec centralite
    de degre")
plt.axis('off')
plt.show()
```



graphe généré

FIGURE 2 – Visualisation du graphe de communication avec couleurs et tailles proportionnelles à la centralité de degré.

Cette analyse permet d'identifier rapidement les individus clés dans un réseau de communications et de comprendre la structure du réseau dans une investigation criminelle.

Exercice 5 : Effet Papillon en Forensique

L'objectif de cette expérience est d'illustrer comment une **petite perturbation temporelle** dans un système de logs peut se propager et affecter la reconstruction complète des événements, suivant le principe de l'effet papillon.

Préparation du système de logs

Nous avons utilisé un jeu de données simulé de 1000 événements corrélés, chaque événement contenant un horodatage précis (`Date et heure`), une source, un identifiant d'événement et une catégorie. Ces logs représentent le type de données qu'un analyste forensique pourrait rencontrer dans un système Windows ou réseau.

Introduction d'une perturbation

Pour modéliser l'effet papillon, un **événement aléatoire** a été sélectionné et son timestamp a été modifié d'une valeur aléatoire comprise entre ±30 secondes. Cette perturbation représente un petit changement local dans l'ordre des événements, semblable à un décalage horaire ou à une incohérence de journalisation.

Observation de l'impact en cascade

Après modification, nous avons calculé le **décalage temporel** de chaque événement par rapport à l'état initial :

$$\delta(t_i) = \left| t_i^{\text{perturbé}} - t_i^{\text{original}} \right|$$

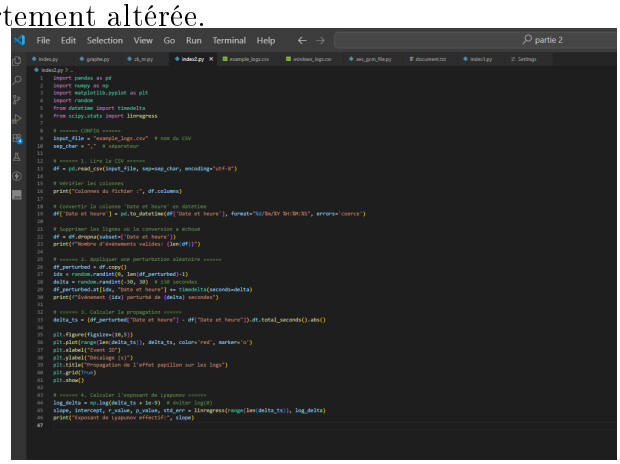
où t_i^{original} est le timestamp initial et $t_i^{\text{perturbé}}$ le timestamp après modification. L'évolution de ce décalage a été tracée sous forme de graphique, montrant comment une petite perturbation peut se propager à travers le système et affecter la reconstruction globale des événements.

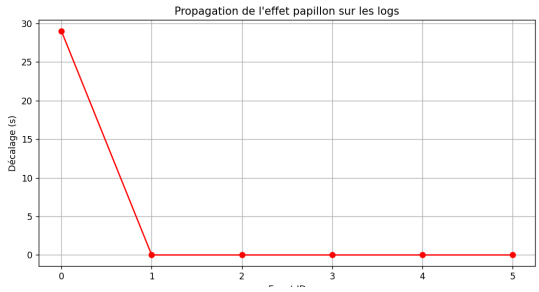
Calcul de l'exposant de Lyapunov

Pour quantifier la sensibilité du système à cette perturbation, nous avons estimé l'exposant de Lyapunov effectif λ à partir de la relation :

$$\delta(t) \approx \delta(0)\, e^{\lambda t}$$

où $\delta(0)$ est le décalage initial. Un exposant de Lyapunov positif indique qu'une petite modification locale peut croître rapidement, entraînant une reconstruction temporelle fortement altérée.





Cette expérience montre que dans les systèmes de logs corrélés, **une micro-perturbation peut avoir un impact significatif**. L'utilisation de l'exposant de Lyapunov permet de quantifier cette sensibilité, confirmant l'importance de la précision temporelle dans l'analyse forensique.

Exercice 6 : Expérience de Pensée Schrödinger Adaptée : Le Fichier Quantique (Q-File)

Nous concevons une transposition de l'expérience du Chat de Schrödinger au domaine numérique pour illustrer le paradoxe de la superposition quantique appliqué aux preuves numériques.

Conception d'une Version Numérique du Chat de Schrödinger

L'analogue quantique du chat est le **Fichier Quantique (Q-File)**, dont l'état binaire (Présent ou Effacé) est lié à l'état d'un qubit dans un système informatique quantique isolé.

- **Le Qubit d'Indétermination (q_i) :** Un qubit initialisé dans une superposition égale : $|\psi_i\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.
- **Le Qubit d'État du Fichier (q_f) :** L'état binaire du fichier ($|0\rangle \rightarrow$ Présent, $|1\rangle \rightarrow$ Effacé) est intriqué avec q_i.
- **L'Opération d'Intrication (Porte CNOT) :** L'application d'une porte CNOT intrique les deux qubits, créant l'état superposé du Q-File (l'équivalent de la fonction d'onde du chat) :

$$|\Psi\rangle_{\text{Q-File}} = \frac{1}{\sqrt{2}}(|0\rangle_{\text{Présent}} + |1\rangle_{\text{Effacé}})$$

  Le fichier est, théoriquement, **simultanément Présent ET Effacé** avant toute mesure.

—

Un Fichier Existe-t-il dans un État Superposé (Présent/Effacé) avant Analyse ?

Théoriquement, **oui**. Tant que le Q-File est maintenu dans un système quantique suffisamment isolé pour prévenir la **décohérence**, il existe dans une superposition cohérente de ses états classiques.

L'acte de l'analyse forensique, qui est une **mesure** ($\hat{M}$), provoque l'**effondrement de la fonction d'onde** :

$$\hat{M}|\Psi\rangle_{\text{Q-File}} \longrightarrow \begin{cases} |\text{Présent}\rangle & \text{avec } P = 50\% \\ |\text{Effacé}\rangle & \text{avec } P = 50\% \end{cases}$$

L'expert légiste ne découvre pas l'état passé ; il **crée** l'état final de la preuve.

—

Quel Impact sur la Notion de Preuve « Certaine » en Justice ?

La superposition quantique introduit l'**incertitude intrinsèque** au cœur de la preuve, remettant en cause le principe de **réalité objective** du droit :

- **Ambiguïté de l'Intention :** Comment prouver l'intention criminelle d'effacer ou de dissimuler un fichier si le statut « Effacé » n'est apparu qu'au moment de l'enquête (la mesure) ? Le défendeur pourrait argumenter que le fichier était dans un état *indéterminé* au moment des faits.

— **Relativité de la Preuve :** La preuve n'est plus une vérité objective et rétroactive, mais une **réalité relative à l'observation**. Le standard de preuve absolue (« au-delà de tout doute raisonnable ») devient paradoxal face à une preuve qui est, par nature, probabiliste avant d'être fixée.
— **Délai de Fixation de l'État :** La date des faits pourrait être considérée comme distincte de la date de la « fixation » de la preuve, ouvrant la voie à des contestations sur l'intégrité de la chaîne de conservation si l'observation n'est pas parfaite.
—

Rédigez un Protocole d'Observation Minimisant l'Effet sur le Système

L'objectif est d'extraire des informations sur l'état du Q-File tout en minimisant l'effet d'effondrement de la fonction d'onde, typiquement en utilisant des techniques de **Mesure Faible (*Weak Measurement*)**.

Protocole d'Analyse Légale Quantique (PAQL)

1. **Isolation Maximale (Prévention de la Décohérence) :**
   — Maintenir le Q-File et son système de qubits associé dans un **environnement ultra-isolé** (cryogénie, vide poussé, blindage électromagnétique) pour retarder au maximum toute interaction environnementale, source de décohérence non désirée.
2. **Mesure Faible (*Weak Measurement*) :**
   — Effectuer des mesures répétées en utilisant une **sonde de faible énergie** (ex. : une impulsion laser ultra-faible) qui ne couple que très faiblement l'appareil de mesure avec le qubit.
   — Chaque impulsion fournit une information **partielle et bruitée**, ne provoquant pas un effondrement immédiat.
   — **Agrégation Statistique :** Les résultats des mesures faibles répétées sont moyennés pour estimer la **valeur d'attente** de l'état du Q-File ($\langle\Psi|\hat{M}|\Psi\rangle$) sans détruire totalement la superposition pour d'autres tests.
3. **Mesure Indirecte par Intrication (Analogue de l'Ami de Wigner) :**
   — Utiliser un **qubit auxiliaire** (*le Qubit-Ami*) pour s'intriquer avec le Q-File avant la mesure.
   — Au lieu de mesurer directement le Q-File, l'expert mesure l'état de ce Qubit-Ami. Tant que le Qubit-Ami n'est pas mesuré par l'observateur classique, la superposition est transférée à l'état combiné du Q-File et du Qubit-Ami, **retardant l'effondrement définitif**.
4. **Documentation de la Mesure Forte :** L'acte final de mesure forte (la lecture binaire et définitive de l'état "Présent" ou "Effacé") doit être documenté comme l'**acte d'effondrement** qui a fixé l'état de la preuve pour le tribunal, reconnaissant ainsi que l'expert est l'opérateur qui a forcé la réduction quantique.

Exercice 7 : Calculs sur la Sphère de Bloch

État du Qubit et Angles

L'état du qubit est donné par l'expression suivante, correspondant aux angles polaire $\theta = \frac{\pi}{3}$ et azimutal $\phi = \frac{\pi}{4}$:

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle = \cos\left(\frac{\pi}{6}\right)|0\rangle + e^{i\pi/4}\sin\left(\frac{\pi}{6}\right)|1\rangle$$

Calcul des Probabilités de Mesure $P(0)$ et $P(1)$

Les probabilités d'obtenir les états $|0\rangle$ ou $|1\rangle$ lors d'une mesure dans la base computationnelle (axe Z) sont données par le carré du module des amplitudes de probabilité.

1. **Probabilité d'obtenir $|0\rangle$:**

$$P(0) = |\langle 0|\psi\rangle|^2 = \left|\cos\left(\frac{\pi}{6}\right)\right|^2 = \left(\frac{\sqrt{3}}{2}\right)^2 = \frac{3}{4} = 0.75$$

2. **Probabilité d'obtenir $|1\rangle$:**

$$P(1) = |\langle 1|\psi\rangle|^2 = \left|e^{i\pi/4}\sin\left(\frac{\pi}{6}\right)\right|^2 = \left|\sin\left(\frac{\pi}{6}\right)\right|^2 = \left(\frac{1}{2}\right)^2 = \frac{1}{4} = 0.25$$

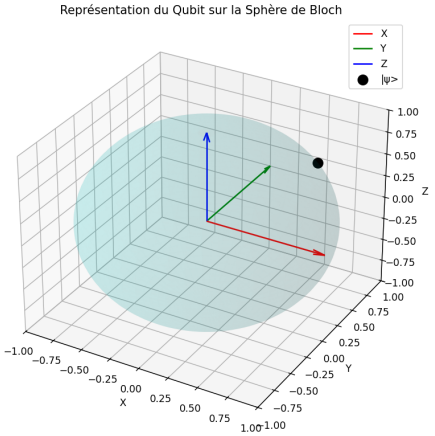
Vérification : $P(0) + P(1) = 0.75 + 0.25 = 1$.

Représentation Graphique sur la Sphère de Bloch

L'état est représenté par le vecteur $\vec{r}$ de coordonnées sphériques $(\theta, \phi) = (\frac{\pi}{3}, \frac{\pi}{4})$.

— L'angle polaire $\theta = \frac{\pi}{3}$ (60°) indique que le vecteur est plus proche du pôle nord ($|0\rangle$) que de l'équateur, ce qui correspond à $P(0) > P(1)$.

— L'angle azimutal $\phi = \frac{\pi}{4}$ (45°) place le vecteur dans le quadrant positif du plan XY (entre l'axe X et l'axe Y).



Impact sur un Système de Preuve Quantique

L'utilisation de cet état $|\psi\rangle$ dans un système de preuve quantique (comme l'intégrité d'un Q-File) aurait deux impacts majeurs :

1. **Perte de Certitude Binaire :** La preuve n'est pas absolue mais probabiliste. Au lieu d'avoir $P(\text{Intacte}) = 1$, on a $P(\text{Intacte}) = 75\%$. La mesure finale réduit la preuve à un état binaire, détruisant l'information probabiliste initiale.
2. **Détection d'Interférence par Phase :** La présence de la phase $\phi = \frac{\pi}{4}$ indique une **cohérence quantique**. Toute tentative de falsification ou altération de la preuve se manifesterait par une **décohérence** (perte de la phase ϕ), agissant comme une signature physique et détectable d'une perturbation.

Exercice 8 : Analyse du Théorème de Non-Clonage

Explication du Théorème de Non-Clonage

Le **Théorème de Non-Clonage (*No-Cloning Theorem*)** est un principe fondamental qui énonce qu'il est impossible de construire un appareil capable de créer une copie parfaite et indépendante d'un état quantique arbitraire et inconnu.

Ce théorème découle de la nature **linéaire** des opérations quantiques unitaires (évolutions temporelles).

1. **Hypothèse de Clonage Parfait :** Supposons qu'un opérateur unitaire U_C puisse parfaitement cloner deux états orthogonaux, $|0\rangle$ et $|1\rangle$, sur un état auxiliaire vierge $|A\rangle$:
$$U_C(|0\rangle \otimes |A\rangle) = |0\rangle \otimes |0\rangle$$
$$U_C(|1\rangle \otimes |A\rangle) = |1\rangle \otimes |1\rangle$$
2. **Violation par Superposition :** Considérons l'état superposé $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$.
   — Par la **linéarité** de U_C, le résultat de l'opération est :
$$U_C(|\psi\rangle \otimes |A\rangle) = \alpha(|0\rangle \otimes |0\rangle) + \beta(|1\rangle \otimes |1\rangle)$$
   — Le résultat attendu pour un **clonage parfait** est :
$$|\psi\rangle \otimes |\psi\rangle = \alpha^2(|00\rangle) + \alpha\beta(|01\rangle) + \beta\alpha(|10\rangle) + \beta^2(|11\rangle)$$
3. **Conclusion :** Les deux résultats sont différents (sauf cas triviaux). L'absence des termes croisés ($\alpha\beta|01\rangle$ et $\beta\alpha|10\rangle$) dans le résultat linéaire prouve qu'un opérateur de clonage parfait ne peut pas être unitaire, et donc n'est pas réalisable physiquement.

Implications pour la Conservation des Preuves Quantiques

Le Théorème de Non-Clonage est une épée à double tranchant pour la gestion des preuves quantiques :

— **Avantage Sécuritaire :** Il **garantit l'authenticité**. Un contrefacteur ne peut pas intercepter et copier une preuve quantique (comme le Q-File de l'exercice précédent) pour la modifier et la réintroduire dans le système. Toute tentative de copie ou de mesure altère irréversiblement l'original.

— **Inconvénient Logistique (Fragilité) :** Il **interdit la sauvegarde**. Il est impossible de créer une « copie de sécurité » parfaite d'une preuve quantique. Si la preuve originale est perdue ou détruite (par décohérence ou erreur de manipulation), elle est définitivement perdue.

Alternative Utilisant le Protocole ZK-NR

Puisque la copie est impossible, la solution de conservation et de vérification réside dans la preuve à divulgation nulle.

Nous proposons d'utiliser un protocole de **Preuve à Divulgation Nulle sans Répudiation (ZK-NR : *Zero-Knowledge No-Repudiation*)**, où la validité de la preuve est vérifiée sans jamais révéler (ni effondrer) son état quantique parfait.

1. **Principe :** Le **Prover** (détenteur de la preuve) prouve au **Verifier** (auditeur/-tribunal) qu'il possède un état $|\psi\rangle$ ayant une propriété spécifique, sans révéler les amplitudes α et β de l'état.
2. **Mécanisme :** Le Prover intrique la preuve $|\psi\rangle$ avec des qubits auxiliaires. Le Verifier envoie des requêtes de vérification sous forme de portes quantiques aléatoires. Le Prover effectue l'opération et retourne les résultats de **mesures faibles** partielles.
3. **Résultat :** Le Verifier peut inférer la validité de la structure quantique de la preuve avec une probabilité très élevée. L'état $|\psi\rangle$ reste non mesuré dans sa superposition, assurant sa conservation et son intégrité maximale.

Exercice 9 : Formalisation mathématique du paradoxe

Définitions

On considère trois grandeurs normalisées sur l'intervalle $[0, 1]$:
— A : **Adéquation / Acceptation**, probabilité qu'un agent rationnel accepte une preuve correcte.
— C : **Certitude / Complétude**, robustesse face aux contre-exemples.
— O : **Opacité / Confidentialité**, degré de préservation de l'information (zéro-connaissance).

Systèmes étudiés

Nous distinguons trois familles de systèmes de preuve :

1. **Classique formel (Hilbert / Coq-like)** : $A \approx 0.98,\ C \approx 0.95,\ O \approx 0.05$.
2. **Probabiliste / interactif (PCP / IP)** : $A \approx 0.90,\ C \approx 0.85,\ O \approx 0.20$.
3. **Zero-Knowledge (ZK-SNARK / ZK-STARK)** : $A \approx 0.88,\ C \approx 0.80,\ O \approx 0.92$.

Inégalité fondamentale

On pose

$$\delta = 1 - A \cdot C.$$

L'inégalité fondamentale est alors

$$A \cdot C \leq 1 - \delta,$$

ce qui est toujours satisfait par définition de δ.

Relation d'incertitude

De manière analogue au principe d'incertitude en physique quantique, on postule :

$$\Delta A \cdot \Delta C \;\geq\; \frac{\hbar_{\text{num}}}{2},$$

où ΔA et ΔC désignent les écarts-types mesurés de A et C. On en déduit une estimation expérimentale :

$$\hbar_{\text{num}} \;\approx\; 2\,\Delta A \cdot \Delta C.$$

Résultats numériques simulés

À partir d'une simulation Monte-Carlo ($N = 200$ échantillons par système), on obtient les résultats suivants :

Système	$\bar{A}$	$\bar{C}$	O	ΔA	ΔC	$A \cdot C$	δ	$\hbar_{\text{num}}$
Classique formel	0.9798	0.9506	0.05	0.0047	0.0069	0.9314	0.0686	6.4×10^{-5}
Probabiliste inter.	0.8983	0.8503	0.20	0.0199	0.0306	0.7638	0.2362	1.22×10^{-3}
Zero-Knowledge	0.8813	0.8020	0.92	0.0096	0.0155	0.7068	0.2932	2.95×10^{-4}

TABLE 1 – Résultats simulés pour les trois systèmes de preuve : moyennes, incertitudes et estimation de $\hbar_{\text{num}}$.

Interprétation

— Les systèmes classiques formels présentent une très faible incertitude ($\hbar_{\text{num}}$ quasi nul).
— Les systèmes probabilistes interactifs montrent une incertitude plus grande, traduite par une valeur de $\hbar_{\text{num}}$ plus élevée.
— Les systèmes Zero-Knowledge se situent entre les deux : forte opacité O, mais incertitude modérée.

Exercice 10 : Implémentation simplifiée : ZK-NR (PoC)

Listing 2 – Version réduite du protocole ZK-NR simulé en Python

```
import hashlib

def H(data: bytes) -> str:
return hashlib.sha256(data).hexdigest()

class MerkleTree:
def commit(self, data: bytes):
```

```
        return {"leaf": H(data), "root": H(data)}

        class STARKProver:
        def prove(self, statement: str, witness: bytes, commitment:
            dict):
        return H((statement + commitment["root"]).encode() +
           witness)

        class MockSigner:
        def __init__(self, i): self.id = i
        def sign(self, message: bytes): return H(f"s{self.id}|".
           encode() + message)

        class ThresholdBLS:
        def __init__(self, threshold=3):
        self.signers = [MockSigner(i) for i in range(5)]
        self.threshold = threshold
        def combine(self, sigs): return H("".join(sorted(sigs)).
           encode())

        class DilithiumSigner:
        def sign(self, message: bytes): return H(b"Dilithium|" +
           message)

        class ZK_NR_Protocol:
        def __init__(self):
        self.commitment_tree = MerkleTree()
        self.stark_prover = STARKProver()
        self.bls = ThresholdBLS()
        self.dilithium = DilithiumSigner()

        def create_attestation(self, document: bytes, metadata:
           dict):
        commitment = self.commitment_tree.commit(document)
        zk_proof = self.stark_prover.prove("I␣know␣D␣with␣hash␣H",
           document, commitment)
        sigs = [s.sign(commitment["root"].encode()) for s in self.
           bls.signers[:3]]
        threshold_sig = self.bls.combine(sigs)
        auth_sig = self.dilithium.sign((zk_proof + threshold_sig).
           encode())
        return {
                "commitment": commitment, "zk_proof": zk_proof,
                "threshold_signature": threshold_sig, "
                   pq_authentication": auth_sig,
                "metadata": metadata
        }

        # Exemple
        proto = ZK_NR_Protocol()
        doc = b"Contrat␣important"
```

```
        att = proto.create_attestation(doc, {"id": "contrat-2025"})
        print(att)
```