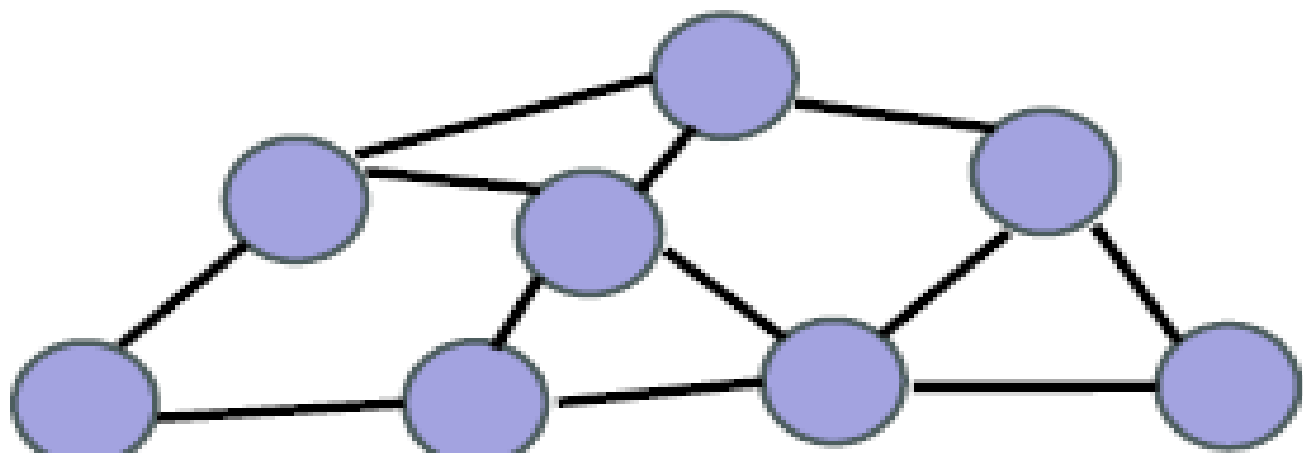


分散量子コンピューティングの最適資源割当

1. 研究背景

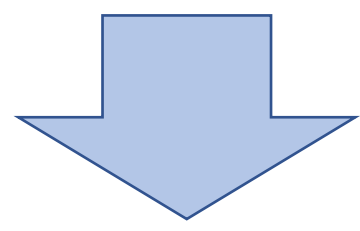
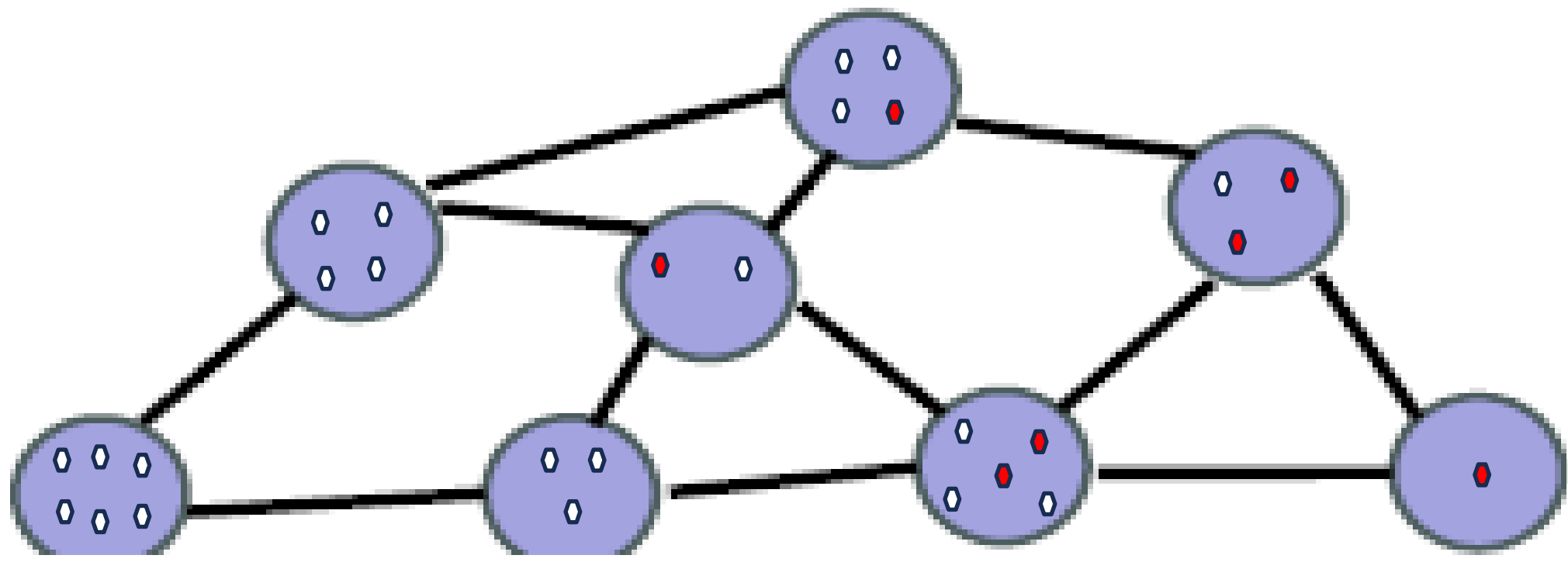
■ 量子ネットワーク

- 量子ビットを転送するネットワーク
- 古典的な「データ転送」とは根本的に異なる原理で転送
- 「量子テレポーテーション」を用いた量子ビットの転送



■ 分散量子コンピューティング (DQC)

- 1台の量子コンピュータの性能には限りがある
- 大規模な計算を複数の量子コンピュータを用いて実現

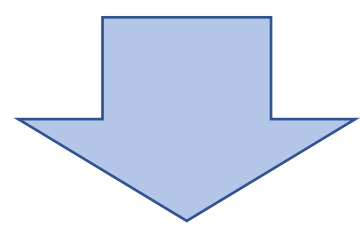


・DQCの実現に向けて
時間によって変動するノード間の通信信頼性に対応
演算の分担や回路の分割をノードの利用状況に応じて調整
ランダムに発生するDQC要求を見越して割り当て

2. 分散量子コンピューティング (DQC)

■ 1台の量子コンピュータ (QC) の性能の限界

- 現在のQCは数10～数100量子ビット数程度の規模
- 実用化するには数1000～数100万の量子ビットが必要



■ DQCのメリット

- スケーラブルな量子計算が可能
- 個々のノードは小型・簡素で良いため、構築しやすい
- 冗長性・フォールトトレランスの実現がしやすい

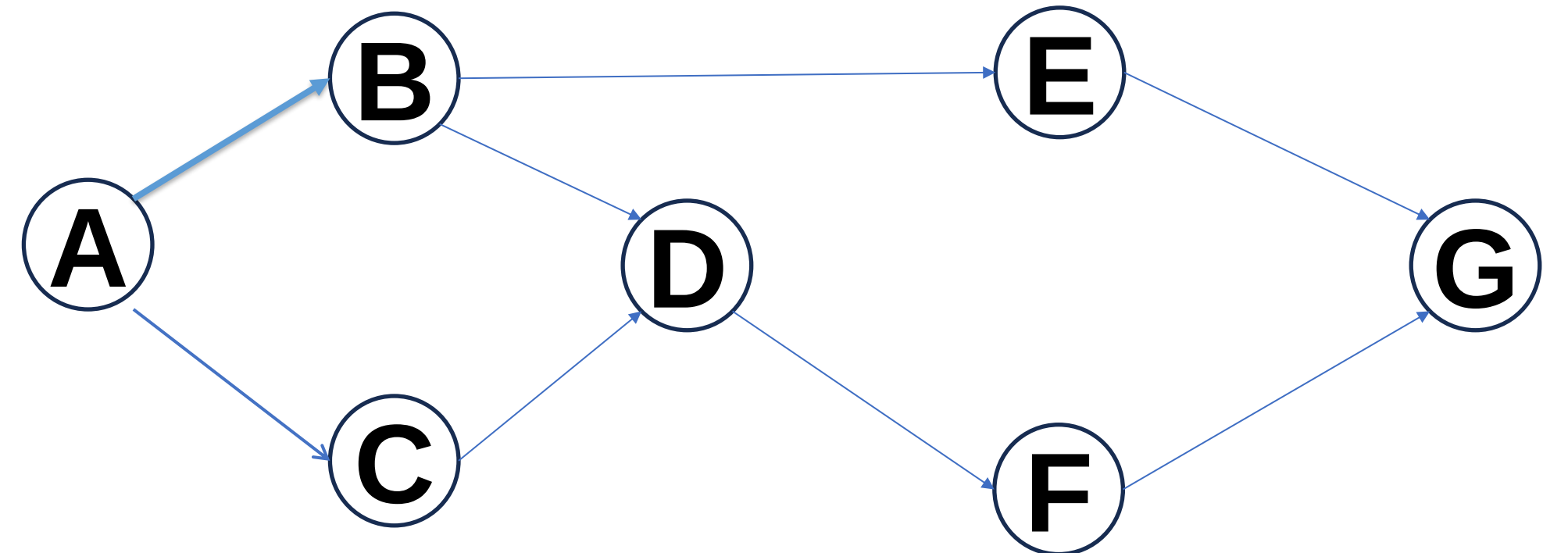
■ ネットワーク構成

- 古典通信 : 測定結果や命令のやり取り
- 量子通信 : エンタングルメントの生成・送信
- エンタングルメント共有 : 複数ノード間で絡み合った状態を共有

3. DAGにおけるDAG

■ DAG (有向非巡回グラフ)

- Directed (有向) : すべての辺に方向がある
- Acyclic (非巡回) : 元のノードに戻るループが存在しない



■ DQCを扱う上で、DAGの重要性

- 量子回路の構造表現
 - ・量子回路は量子ビットに対するゲートの依存関係を持つ
 - ・DAGで表現することで、並列実行可能なゲートと順序依存のゲートを明示
- 分散環境での最適なゲート配置
 - ・複数の量子ノードに回路を分割して割り当てる必要がある
 - ・DAGによって通信回数の最小化や依存関係を保った分割ができる
- トポロジカルソートによるスケジューリング
 - ・DAGのトポロジカル順序を使って各ゲートの実行タイミングを決定できる

■ 既存の研究

- 単一のDAGを対象とした静的な割り当て

4. 提案手法

■ 総コストが最小となるよう動的にDAGを物理QCに割り当てる

■ DQC要求の発生パターンが既知の場合

- DAGによるDQC要求のモデル化
- 物理QCと量子ネットワークのモデル化
- コスト最適化問題の定式化
- メタヒューリスティックな解導出アルゴリズムの検討

■ DQC要求の発生パターンが未知の場合

- 強化学習による資源割当法のモデル化
- DQC要求の生成モデルの作成

■ 評価

- 計算機シミュレーションによる数値評価

5. 今後の予定

■ より適正な評価を行えるよう量子ネットワーク、量子コンピュータについてのさらなる理解

■ DAGによるDQC要求のモデル化