

# リング方式による IP 電話会議のシミュレーション評価

040427500 山中裕司  
渡邊研究室

## 1. はじめに

通信技術の発達や、低価格化などにより、家庭から企業まで IP 電話の普及が進んでいる。それに伴い、企業などでのみ利用されていた多者間通話も手軽に利用できるようになってきている。しかし、現在 IP 電話システムの多くは 1 対 1 の通話がほとんどで、多者間通話システムは特殊なサーバが必要であったり、トラフィックが増加したりするなどの課題がある。我々は音声パケットをリング状に中継させることにより、サーバが不要で、かつトラフィック量を抑えることができるリング方式の IP 電話会議を提案している[1]。本稿では、ns-2 によるシミュレーション評価を行い、この方式が有用であることを証明した。

## 2. 既存技術

既存の多者間通話方式として、大きく分けて中央処理方式と全セッション方式の 2 つがある。中央処理方式はミキシングを担当する特殊なサーバを利用する方法でトラフィックを少なく抑えられるが、端末の増加と共にサーバの負荷が大きくなるという課題がある。全セッション方式は全端末が個別にセッションを張る方式で、特殊なサーバを必要とせず柔軟性が高いが、参加端末が多くなると大量のトラフィックが流れてしまうという課題がある。

## 3. リング方式

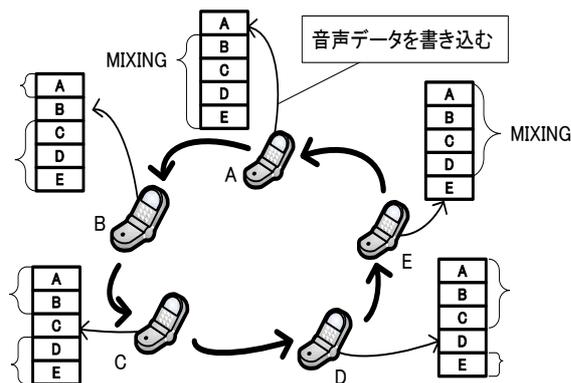


図 1. リング方式の概要

図 1 にリング方式の概要を示す。リング方式では各端末がリレーパケットを次の端末へリング状に回していく。リレーパケットには通話に参加する端末分の領域を用意し、各端末はリレーパケットを受信すると、自分の領域に音声データを書き込み、次の端末へ中継する。その後、自分の領域以外の音声データをミキシングして再生する。

各端末はリング構成を常に把握しておくために、一定時間毎に相互にヘルスチェックを行う。ヘルスチェ

ックパケットが一定時間来ない端末は、無断で会議から離脱したと判断し、リング構成を変更する。

リングを構成する端末の順番は IP アドレスの小さい順とする。これにより同一ネットワークに接続する端末はルータを経由せずに中継され、パケットが無駄な経路を中継することを回避できる。

インターネットなどの信頼性の低いネットワークを介する場合は、パケットロスによる影響を抑えるため、リングの代表端末同士を全セッション方式で結合する。

## 4. ns-2 によるシミュレーション評価

3 つのネットワークをそれぞれルータで繋ぎ、ネットワーク毎に端末を均等に配置し、端末数、音声コーデックを変化させ、全セッション方式とリング方式を比較した。それぞれ 100 秒間の IP 電話会議を行い、ルータを通るパケット数の平均値を測定した。

図 2 にシミュレーション結果を示す。全セッション方式では、人数が増えるごとにルータに大きな負荷を与えているが、リング方式ではパケットを 1 つにまとめ、無駄な経路を通らないため、ルータを通るパケット数は最小限に抑えることができることがわかる。

また、リング方式では音声データを 1 つのパケットにまとめて巡回させるだけであるため、パケット数だけでなく、ネットワーク上のトラフィックも大幅に削減させることが可能である。

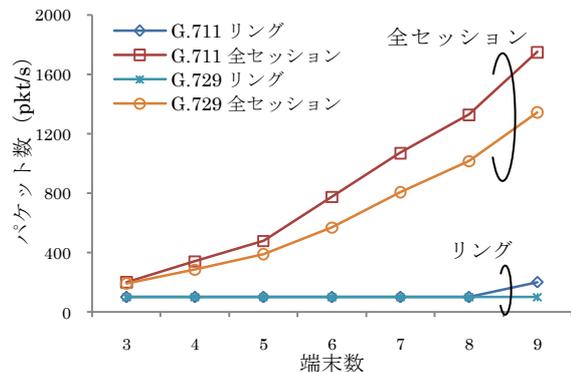


図 2. ルータを通るパケット数

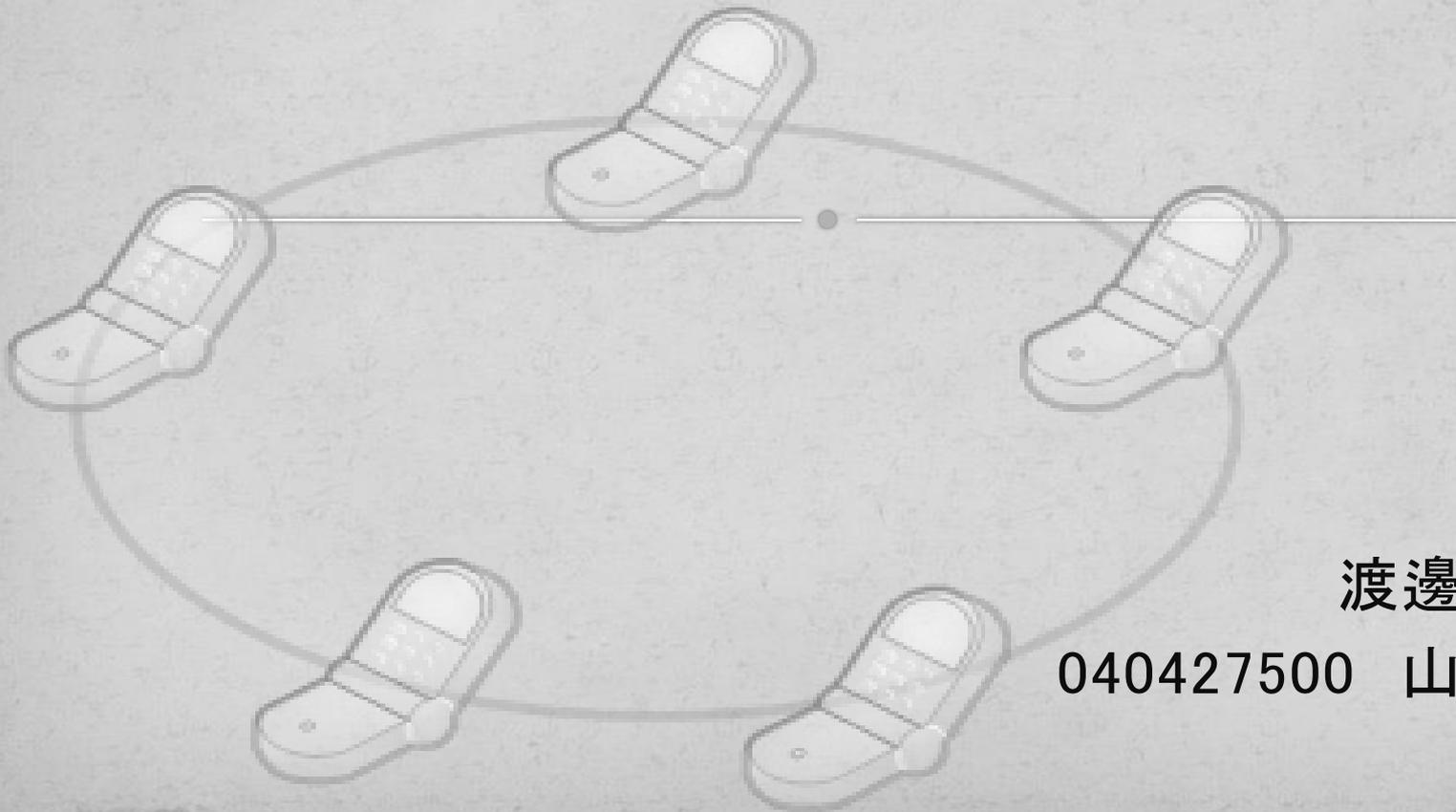
## 5. まとめ

本稿では音声パケットをリング状に中継するリング方式による IP 電話会議を検討した。また、ns-2 によるシミュレーション評価を行った結果、その有効性が明らかとなった。今後はリング方式の実装を行う。

## 参考文献

[1] 陳華龍, 伊藤将志, 渡邊晃: 多者間通話方式の検討, 電子情報通信学会総合大会, Sep.2005.

# リング方式によるIP電話会議 のシミュレーション評価



渡邊研究室

040427500 山Φ裕司

# 多者間通話の背景と課題

## 背景

- 通信技術の発達によるIP電話の実用的な品質保証
- 低価格による家庭から企業までIP電話の普及

多者間通話の利用範囲が大きく広がる

## 課題

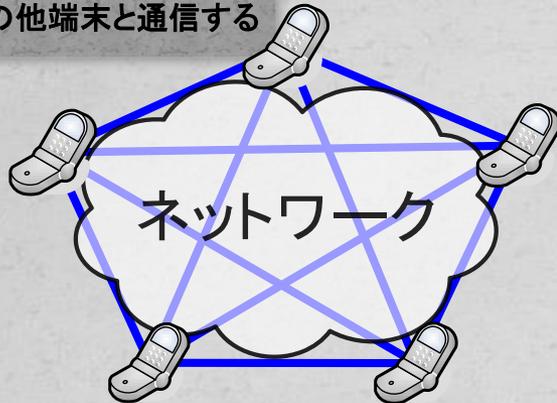
- 多くのシステムが一對一の通話機能に留まっており、多者間通話を実装するシステムは少ない
- 多者間通話の課題
  - 複数端末の通話参加によるネットワークトラヒックの増加
  - 音声をまとめる(ミキシングする)装置にかかる負荷

手軽に利用できる多者間通話はまだ普及しているとは言えない

# 既存方式

## 全セッション方式

各端末がピアツーピア  
で全ての他端末と通信する



### ■ 利点

- 容易に多者間通話を実現できる
- サーバを必要とせず，場所や環境への柔軟性が高い

### ■ 欠点

- 参加端末が多くなると，ネットワークトラフィックが膨大になる
- ルータへの負担が大きくなる

### ■ 利点

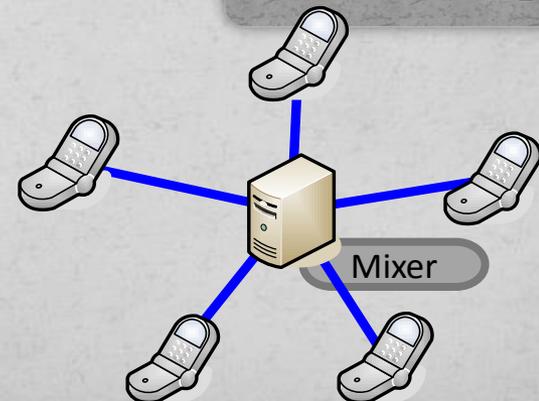
- 各参加端末と中央処理サーバの間のセッションのみであるため，トラフィックは少ない
- ほとんどの作業をサーバが処理するために，制御しやすい

### ■ 欠点

- ミキシング処理のための高性能のサーバを必要とする
- 利用できる端末数が制限される

## 中央処理方式

中央のサーバが音声データの  
ミキシング処理を行う



# リング方式

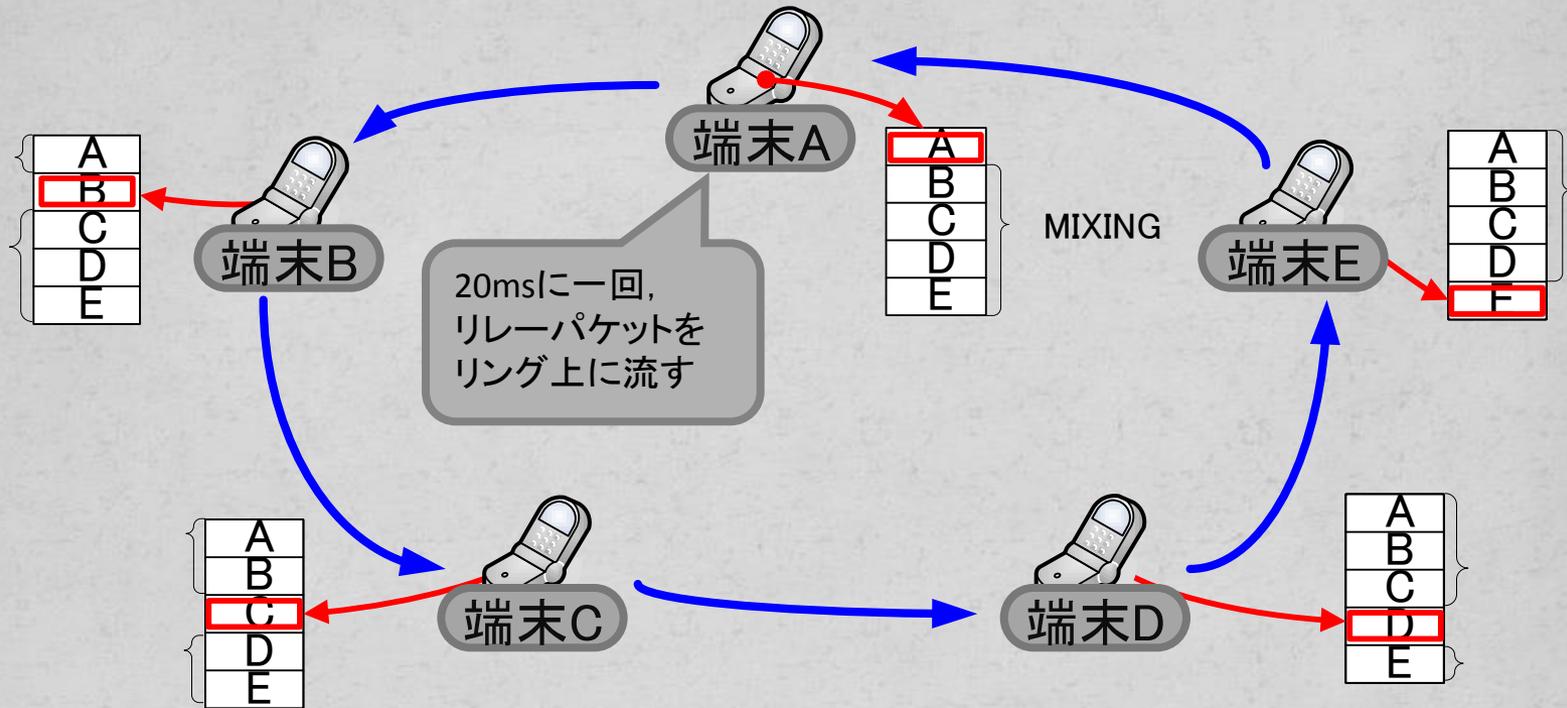
- 特殊なサーバを必要としない
- ネットワーク上のトラヒックを抑える



## リング方式による多者間通話

会議に参加する端末の音声データを1つのパケットに集め、端末間でリング状になるように中継する

# リング方式の概要



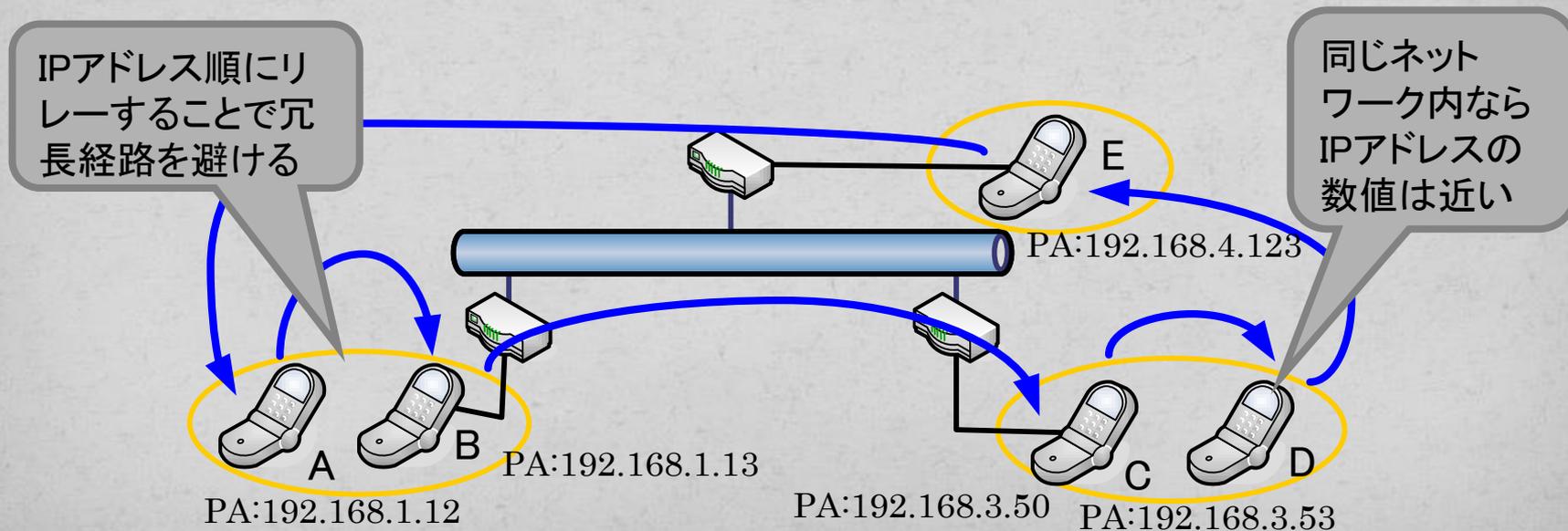
- 代表端末(端末A)は各端末を呼びかけ会議を開始させる
- 端末Aは各端末のデータ領域を持つリレーパケットを生成
- 音声データを自身の領域に書き込み中継する
- 音声の入っている領域をミキシングしてから再生する

# IPアドレスによるリング構造生成

リングはIPアドレス順でリレーするように構成



同一ネットワーク内の端末はリングの順番が隣り合うことになり冗長な経路になることを避けることができる



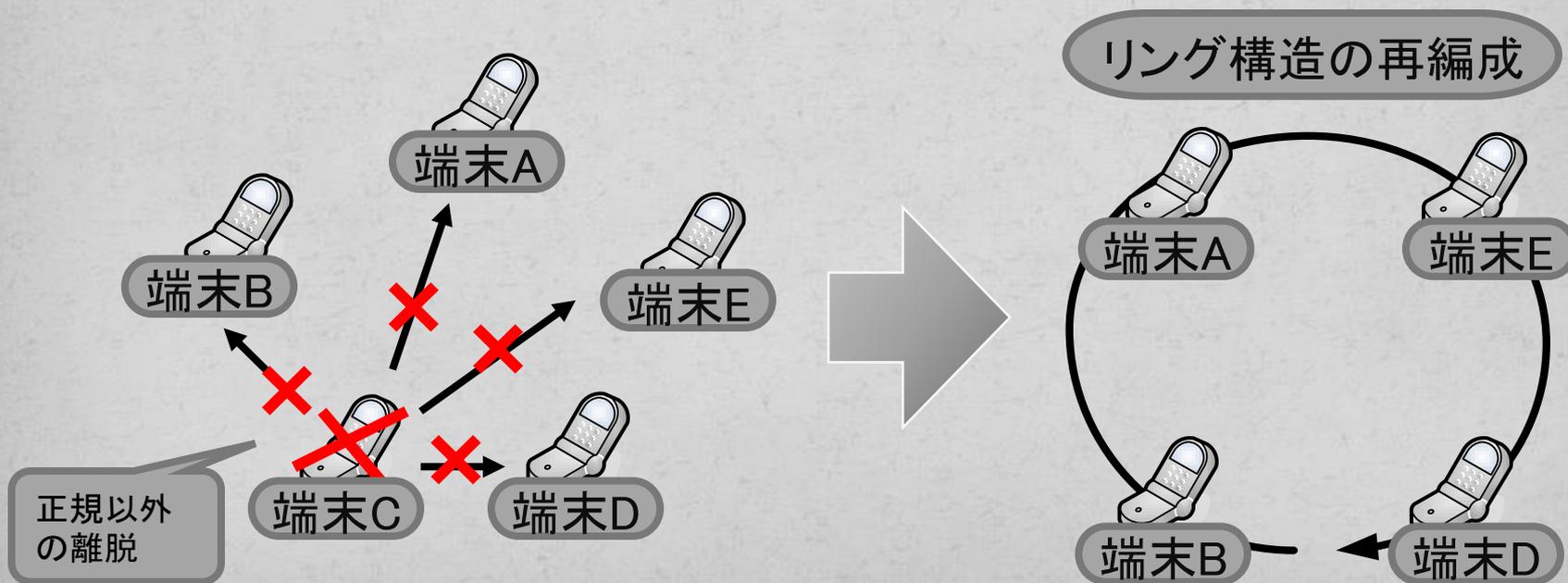
※PA・・・プライベートアドレス

# ヘルスチェック

端末が正規外の離脱をするとリング構造の修正を行う



一定時間毎に他の全ての端末にヘルスチェックを行うことで、各端末がリング状に生きていることを確認する



# ns-2によるシミュレーション評価

- リング方式が有効であることを証明するためにns-2によるシミュレーション評価を実行し、全セッション方式と比較する

- ① ルータを通るパケット数ネットワーク全体のトラフィック量
- ② TCP通信へ与える影響
- ③ 背景負荷をかけたときのパケットロス

企業内でのIP電話会議として利用することを想定

# ns-2による性能評価実験(1)

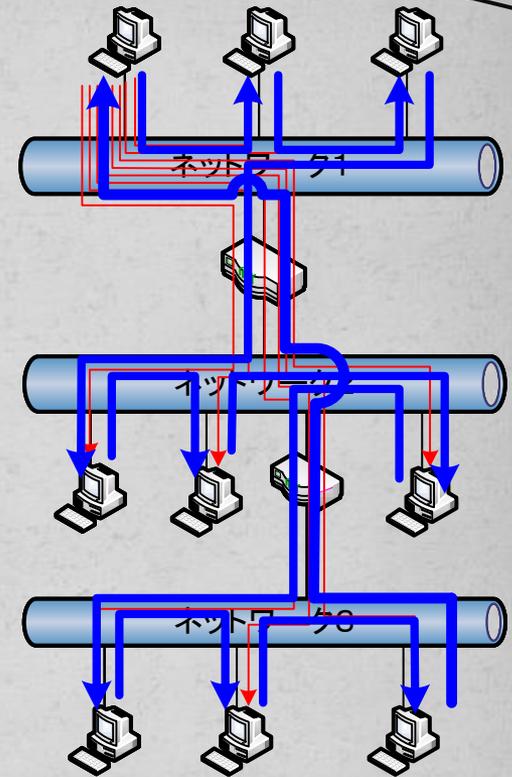
ルータを通るパケット数  
ネットワーク全体のトラヒック量

## ・目的:

両方式のパケット数の違いがルータに与える影響を調べるためにルータを通るパケット数を測定する。また、リング方式は構造上パケット長が長くなるので、それがネットワークにどのような影響を与えるかトラヒック量を測定する。

## ・方法:

端末数、音声コーデックを変化させ、全セッション、リング方式それぞれで100秒間のUDP通信を行い、ルータを通るパケット数、ネットワーク全体のトラヒック量を測る



## ・G.711: (圧縮率 低)

PCM符号化方式で、IP電話サービスで主に使われている

## ・G.726: (圧縮率 中)

ADPCM方式で、ほとんど品質を変えずにデータ量をG.711の半分にできる.PHSサービスで主に使われている

## ・G.729: (圧縮率 高)

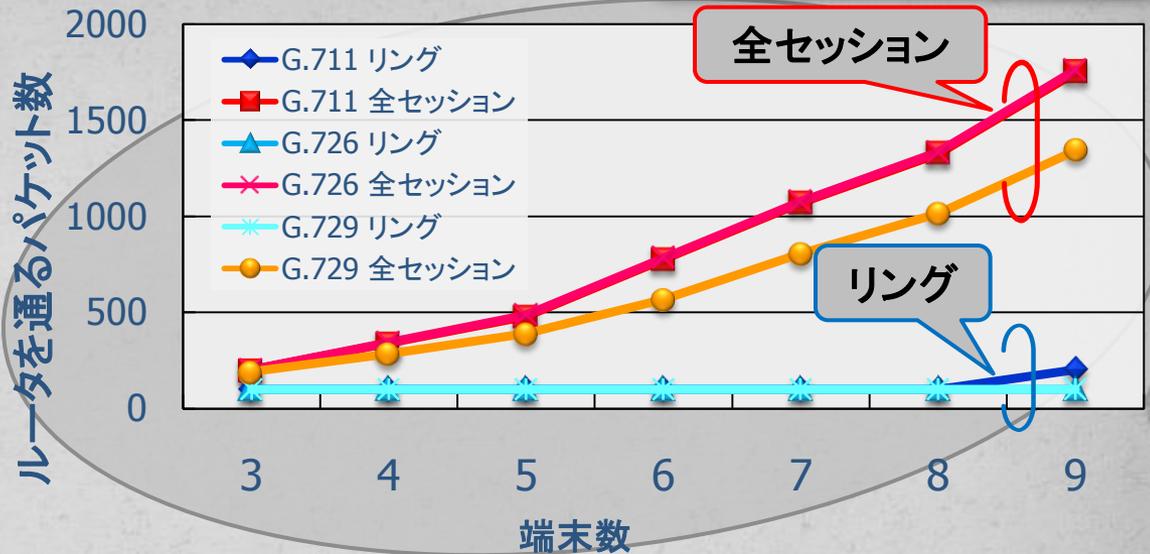
CELP符号化方式で、圧縮率は最も高くなるが、圧縮処理が複雑で演算量が多くなる.携帯電話などで使われる

全セッション・リング端末数: 3~9, ルータ: 2  
帯域幅: 100MB, パケット送出間隔: 20ms  
測定時間: 100sec, LAN: IEEE802.3  
パケットサイズ:  
リング: G.711:  $42+172*n$  (nは端末数)  
          G.726:  $42+92*n$   
          G.729:  $42+32*n$                    bytes  
全セッション: G.711: 214  
                  G.726: 134  
                  G.729: 74                   bytes

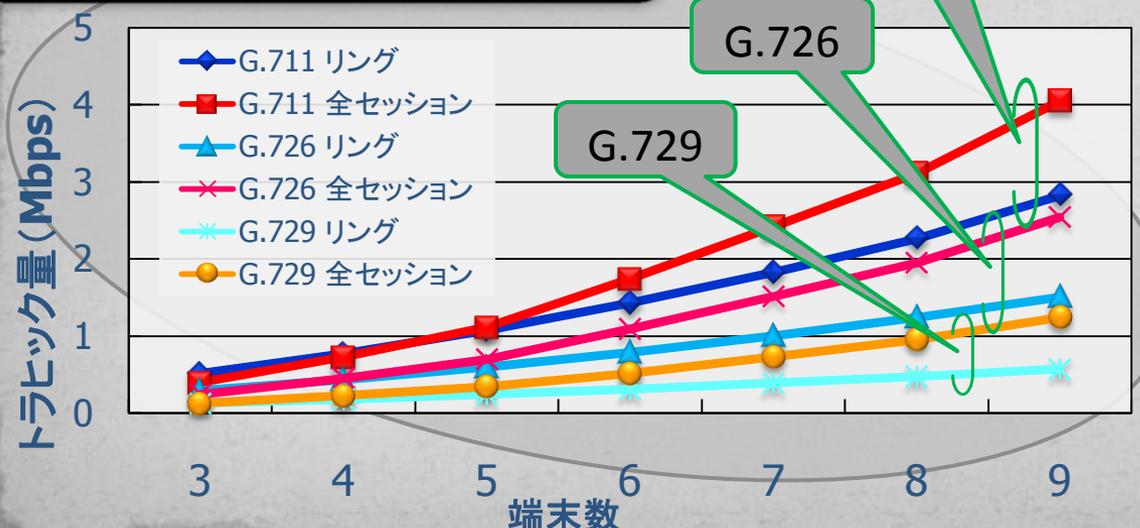
# 性能結果(1)

## ルータを通るパケット数

- 全セッション方式では人数が増えるごとにルータにかかる負荷が大きく増加する
- リング方式ではリレーパケットを使い、冗長な経路をたどらないので、ルータにかかる負荷はほぼ一定



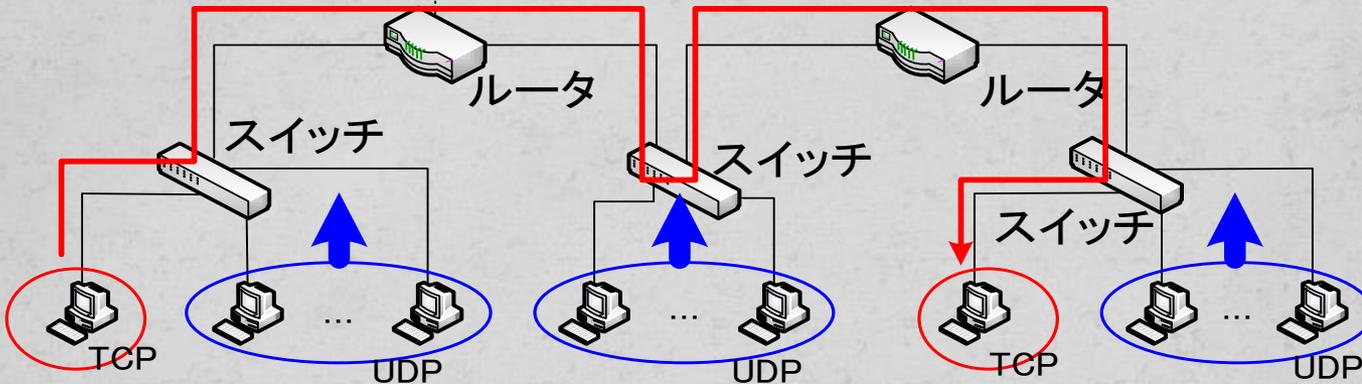
## ネットワーク全体のトラフィック量



- 各々のコーデックに於いてリング方式に有利な結果となっており、その差は人数が増える毎に顕著になっている

# ns-2による性能評価実験(2)

TCP通信へ与える影響



全セッション・リング端末数: 6~30  
TCP通信用端末数: 2,            ルータ: 2  
スイッチングハブ: 3  
帯域幅: 100MB, パケット送出間隔: 20ms  
測定時間: 30sec, LAN: IEEE802.3  
TCPバージョン: New Reno  
パケットサイズ:  
リング: G.711: 42+172\*n (nは端末数)  
          G.726: 42+92\*n  
          G.729: 42+32\*n            bytes  
全セッション: G.711: 214  
                  G.726: 134  
                  G.729: 74        bytes  
TCP: 1000        bytes

## ・目的:

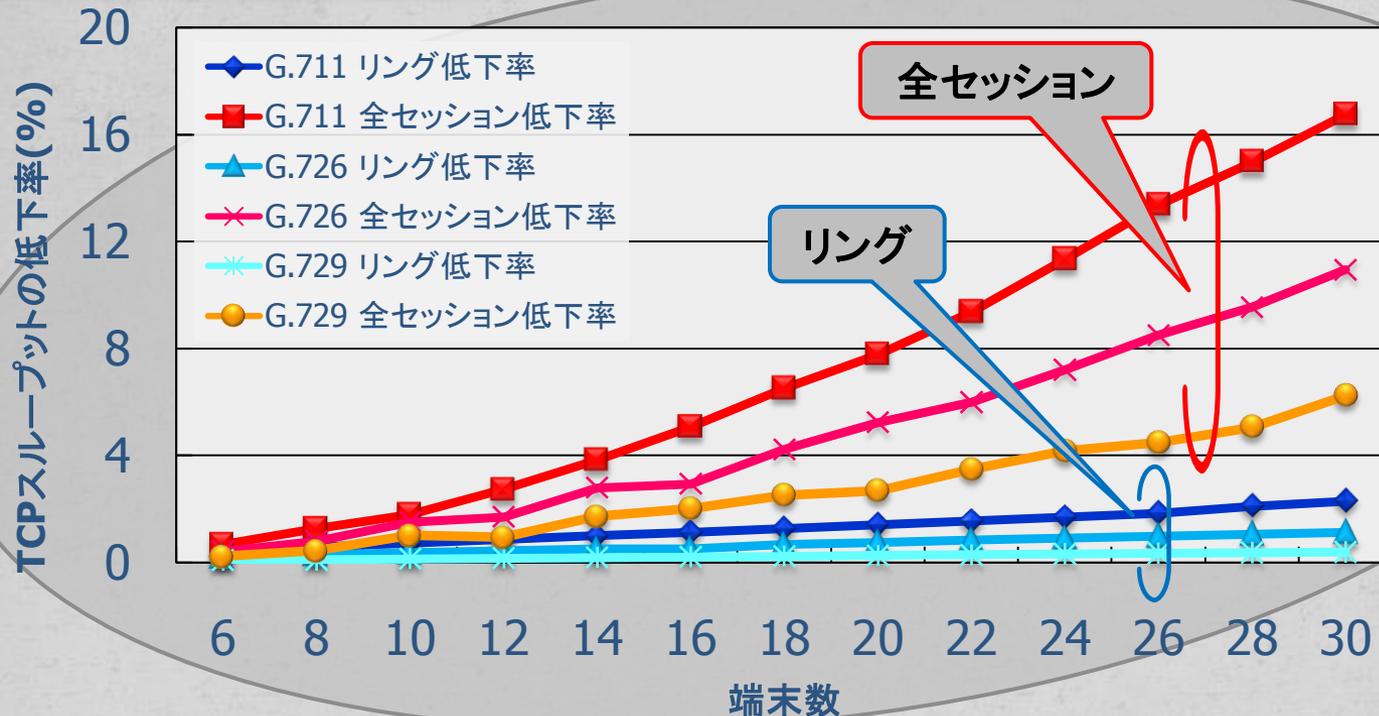
両方式が他の通信に与える影響を調べるため、TCP通信を行う端末のそばで全セッション、リング方式での通信を行い、TCPのスループットがどれだけ低下するか測定する

## ・方法:

ほぼ同時にTCP通信とリング・全セッション方式でのUDP通信を開始し、端末数、音声コーデックを変化させ30秒間の通信を行うことで、TCP通信のみのときと比べ、TCPのスループットがどれだけ低下するかを測定する

# 性能結果(2)

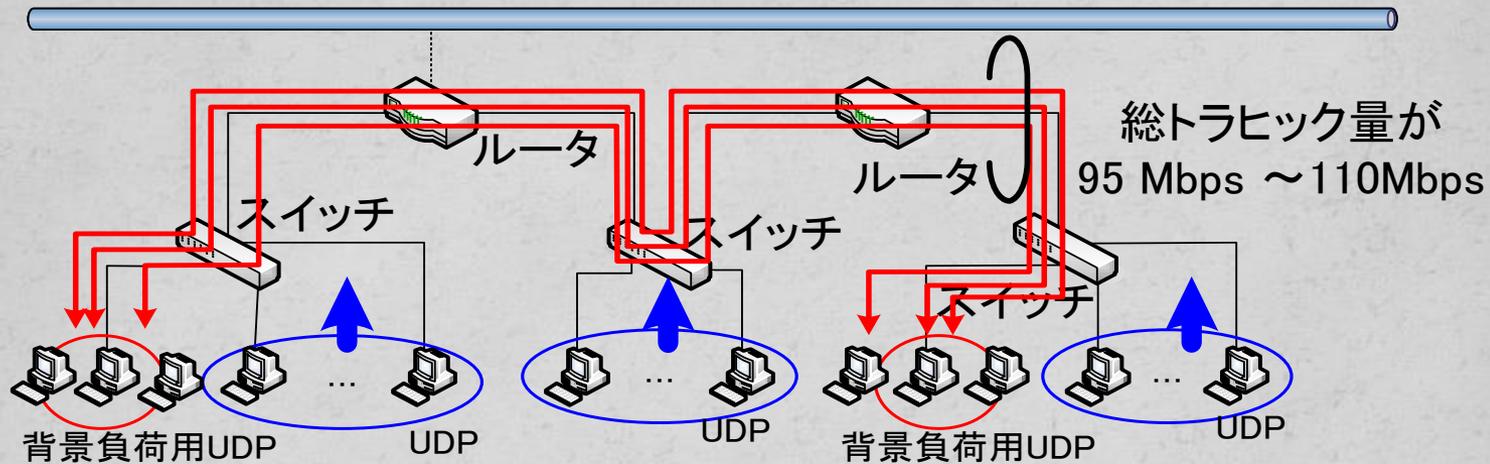
## TCPスループットの低下率



- 全セッション方式では参加人数が増えるごとに発生する大量の packets がルータの大幅なキューイング遅延を引き起こし、TCP通信に大きな影響を与える
- リング方式では音声データを1つのリレーパケットにまとめ、IPアドレス順にリレーされるのでTCP通信にはほとんど影響を与えない

# ns-2による性能評価実験(3)

背景負荷をかけたときのパケットロス



## ・目的:

両方式が他の通信から受ける影響を調べるため、UDP通信を行う端末のそばで全セッション、リング方式での通信を行い、両方式がどれだけパケットロスするかを測定する

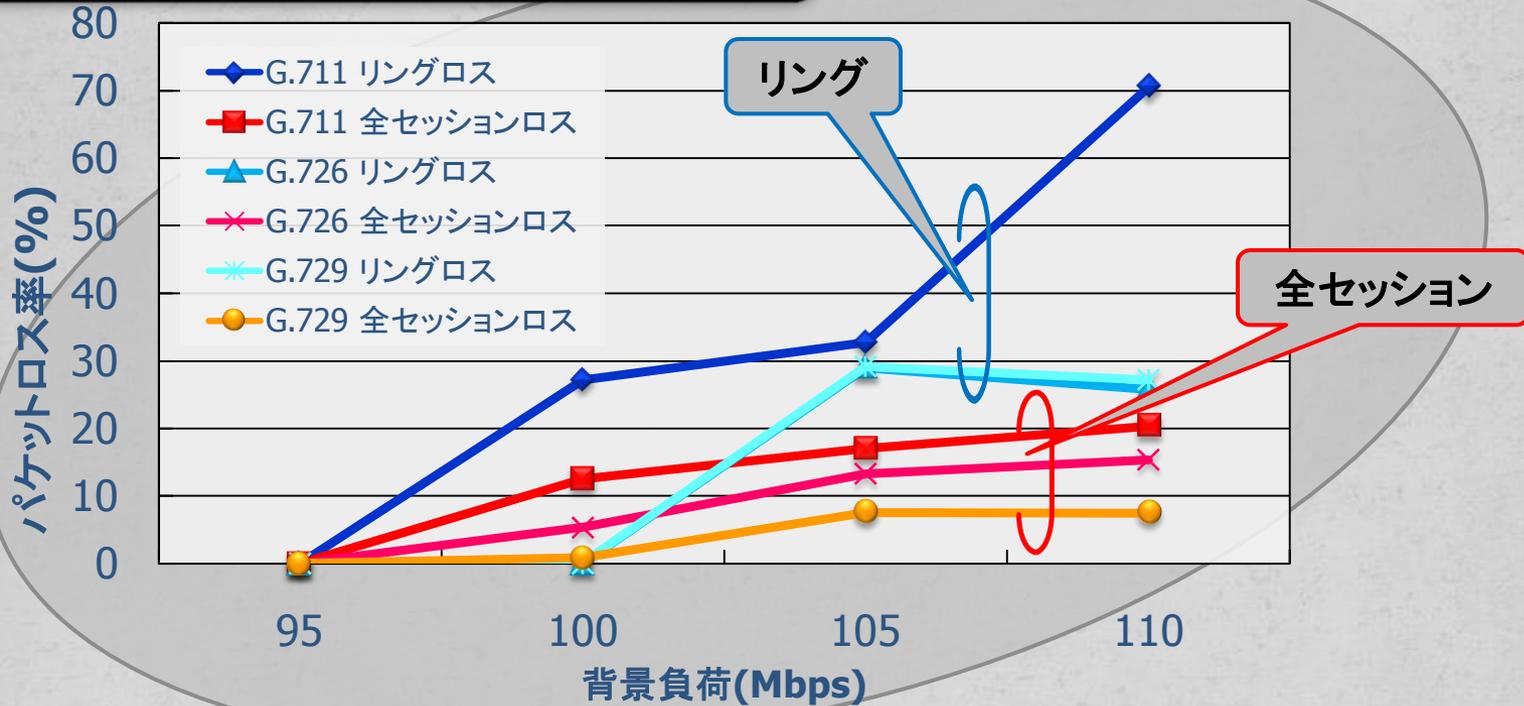
## ・方法:

三対の総トラヒック量が95 Mbps ~ 110MbpsのUDP通信による負荷をかけた状態でリング、全セッション方式によるUDP通信を30秒間測定し、両方式のパケットロス率を測定する

全セッション・リング端末数: 10  
UDP背景負荷用端末数: 6, ルータ: 2  
スイッチングハブ: 3  
帯域幅: 100MB, パケット送出間隔: 20ms  
測定時間: 30sec, LAN: IEEE802.3  
パケットサイズ:  
リング: G.711: 42+172\*n (nは端末数)  
G.726: 42+92\*n  
G.729: 42+32\*n bytes  
全セッション: G.711: 214  
G.726: 134  
G.729: 74 bytes  
UDP: 1000 bytes

# 性能結果(3)

## リング・全セッション方式の packet loss rate



- リング方式では帯域の限界以上の負荷が与えられると、パケットロス率が急に上がる

# 性能結果(4)

	全セッション方式	リング方式
パケット数	×	○
トラヒック	×	○
他通信に与える影響	△	○
パケットロス	○	△

- パケット数・トラヒックに関して有利
- 他通信に与える影響を少なく抑えることができる
- ネットワークが過負荷な状態ではパケットロスにより通信が困難

リング方式は有効である

# まとめ

- リング構造を用いた多者間通話の提案とns-2によるシミュレーション評価
  - リング方式の提案
    - 各端末のデータ領域を持つリレーパケットの生成
    - IPアドレスによるリング構造の生成
    - ヘルスチェックによる相互生存確認
  - ns-2による性能評価実験
    - ルータを通るパケット数, ネットワーク全体のトラフィック量
    - TCP通信へ与える影響
    - 背景負荷をかけたときのパケットロス
- 今後の課題
  - 実装
  - 大規模化(全セッション方式との組み合わせ)

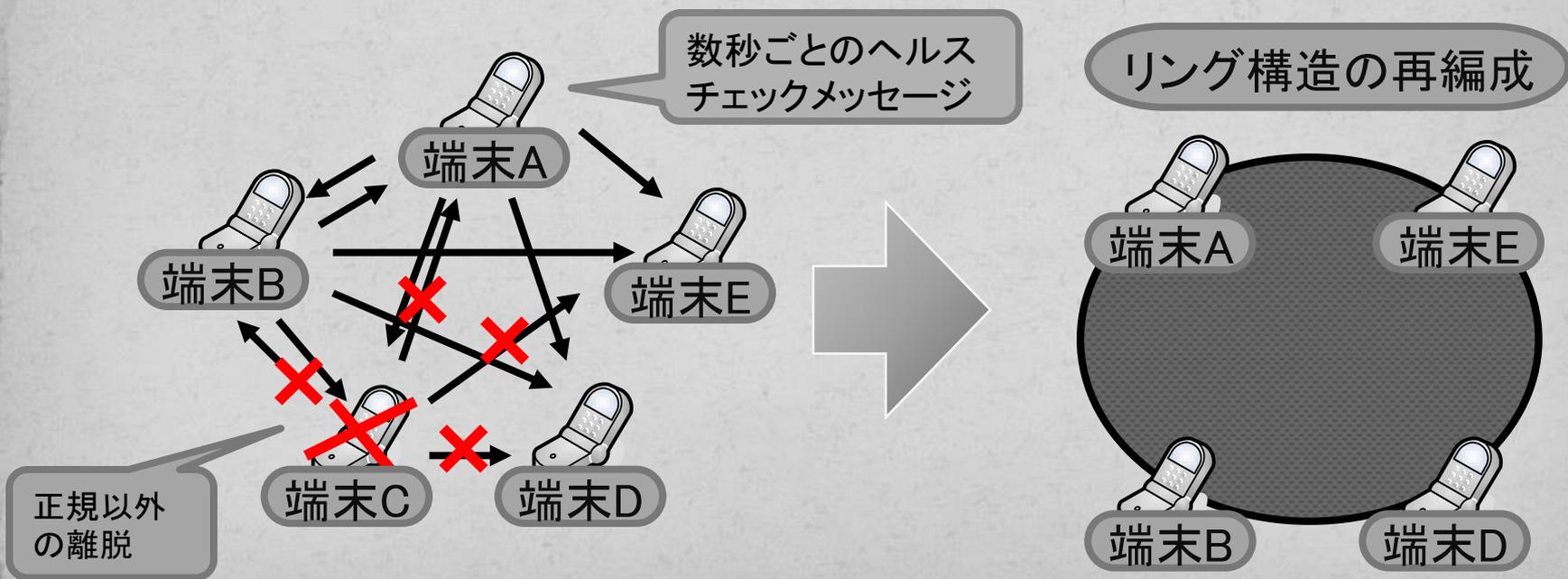
# 補足資料

# ヘルスチェック

端末が離脱するとリング構造の修正を行う

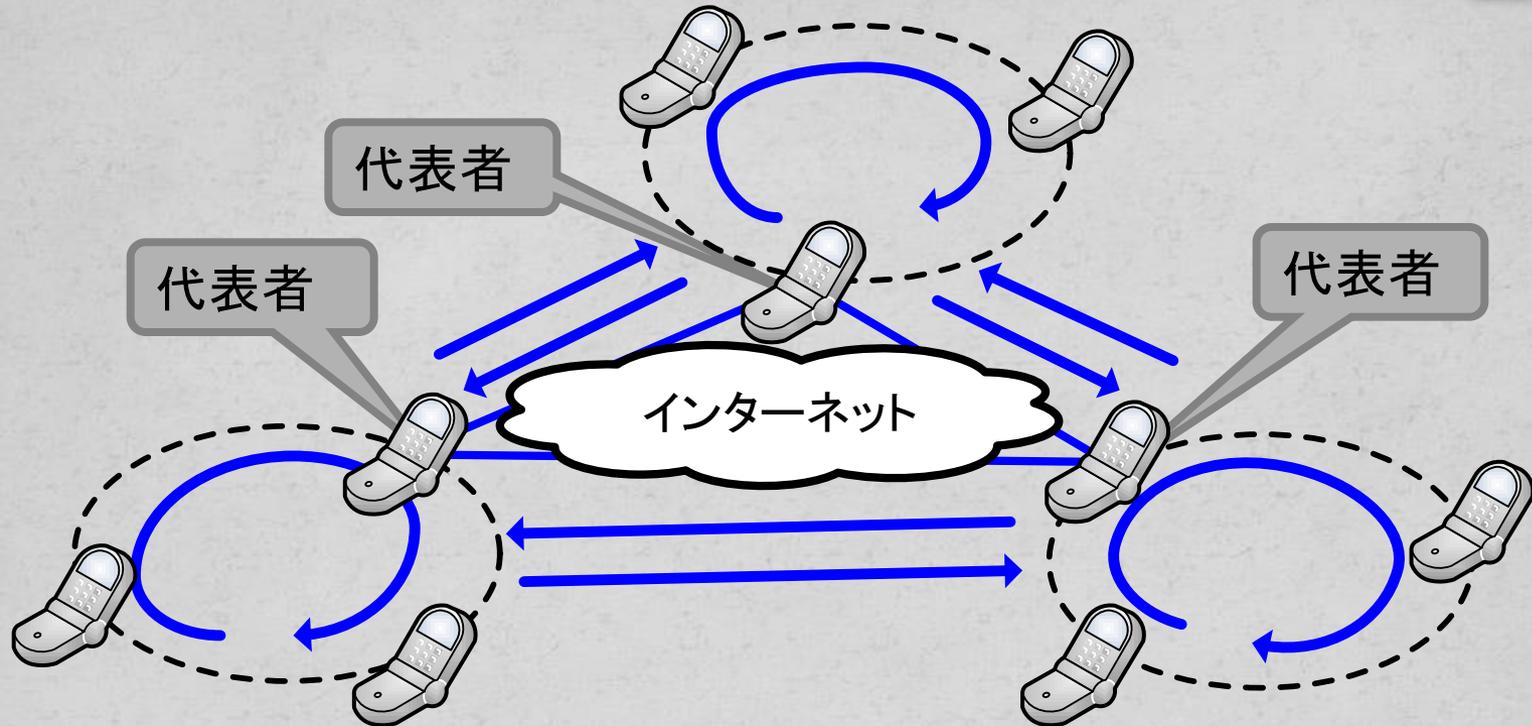


一定時間毎に他の全ての端末にヘルスチェックを行うことで、各端末がリング状に生きていることを確認する



# 全セッション方式との結合

検討



- 代表者はパケットをコピーし、他の代表者へ全セッション方式で中継する
- 他の代表者から受け取ったパケットは、同じネットワークの全端末に中継する

# 既存技術

- Skype
  - 全セッション方式
  - 小規模～中規模
- Windows messenger
  - 全セッション方式
  - 小規模～中規模
- MCU (Multipoint Conference Unit)
  - 中央処理方式
  - 小規模～大規模