

シミュレーションによるネットワーク性能評価システムの診断機能

2F-5

関 義長* 宮木陽一** 岩本真治*

* 日本電気(株)C&C システムインタフェース技術本部, ** 日本電気技術情報システム開発(株)

1 はじめに

近年、ワークステーションの普及や分散化により複数のコンピュータがネットワークを介して利用されるようになってきた。このようなコンピュータネットワークシステムを構築する際に、性能を考慮して設計することは重要なことである。そこで、我々はネットワークの性能を考慮した設計を支援するためのシミュレータ SENSE を開発している [1]。

本稿では、シミュレータ SENSE のシミュレーション結果からネットワークシステムの性能上の問題を発見し、その問題に対する解決案を提示する診断機能について述べる。

2 ネットワークシミュレータ SENSE

SENSE では、グラフィックユーザインタフェース上でノードやネットワークを表すアイコンを使ってネットワークシステムのモデル定義を行なう。シミュレーションの実行結果は、一覧表とグラフで出力される。

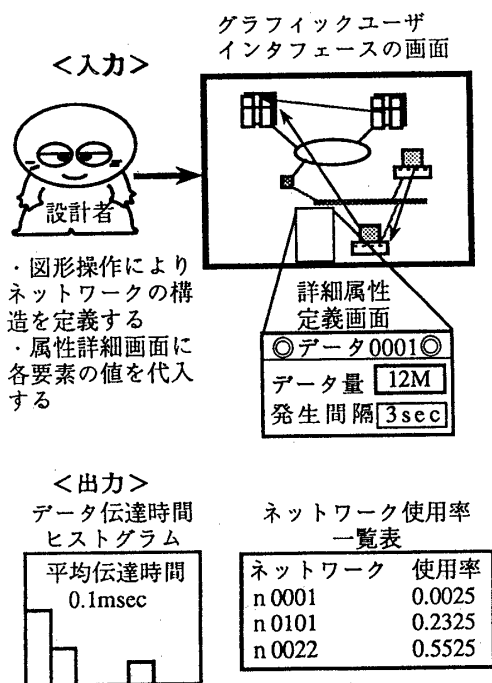


図 1: シミュレータ SENSE

SENSE の出力機能の1つとして現在、診断機能の開発を行なっている。診断機能は、シミュレーションの結果より性能上問題のある所をネットワークシステムのモデル図の中で図示し、その原因及びそれを解決する案を提示する機能である (図2参照)。

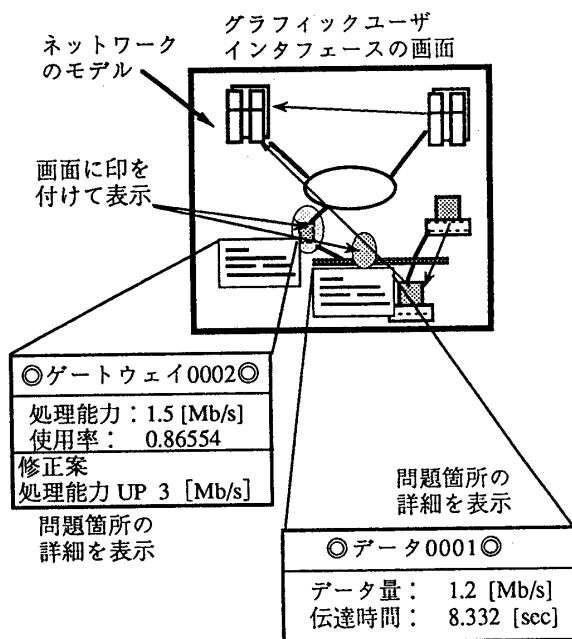


図 2: 問題箇所の表示

3 診断機能

診断機能の構成を図3に示す。シミュレータのユーザは、ネットワークシステムのモデル定義を行なう時に、ユーザの望む性能についての要求を入力する。シミュレーションの解析結果は表示のためにユーザインタフェース、あるいは診断のために診断部に渡される。診断部では、ユーザのネットワークに対する要求とシミュレーション結果に基づき、性能上問題となるネットワークの構成要素を発見し、それをグラフエディタ上で印をつけて表示する。さらに、発見された問題を解決するために修正ルールを用いて、最初にユーザが作成したモデルに対してネットワークの容量やノードの処理能力等の増減を提案する修正案を求めて表示する。

Diagnosis Function of a Network System Performance Evaluation using Simulation

Yoshinaga SEKI* and Youichi MIYAKI** and Shinji IWAMOTO*

*C&C Systems Interface Eng. Lab., NEC Corporation

**NEC Scientific Information System Development Ltd.

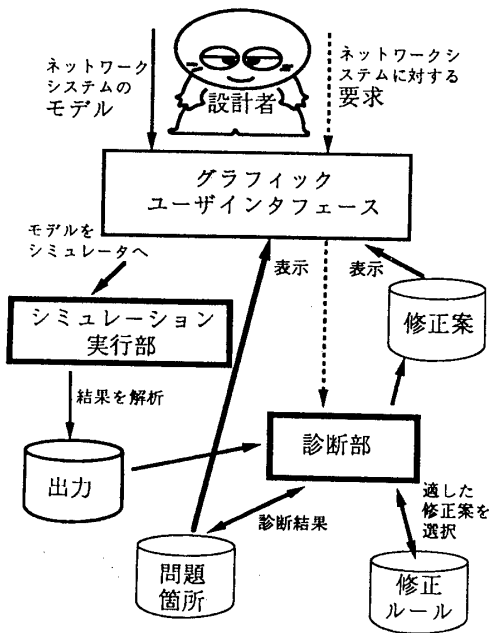


図 3: 診断機能の構成

4 診断方式

診断は、ネットワークの性能上の問題を発見する問題発見フェーズと、問題を解決するための修正案の提示を行なうための問題解決フェーズからなっている。図 4 に診断例を示す。

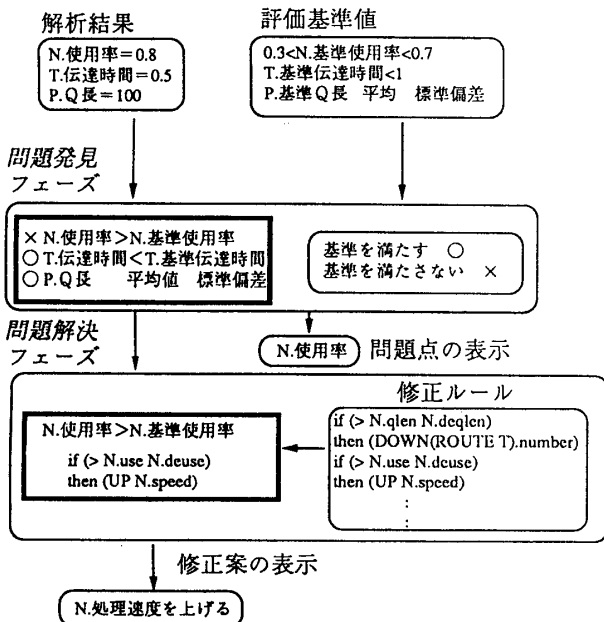


図 4: 診断の例

問題発見フェーズ

ユーザのネットワークの性能に対する要求を、診断するための評価の基準として、シミュレーションの解析結果の診断を行ない、要求を満たしていないネットワークの構成要素を問題有りとして抽出する。

問題解決フェーズ

問題解決フェーズにおいては、問題発見フェーズで発見された問題に対してとるべき対策を修正ルールより選択する。修正ルールには、if-then の形式で、性能上問題となった症状とそれに対する対策が示してある。

これは、if (条件式の and 結合) then (修正関数 修正量 要素) の形式を持つ。条件式は (比較演算 要素 評価基準値) である。要素は、ノード、ネットワーク、トランザクションの識別子と属性で表現される。以下に修正ルールの例を示す。

```

1 if (> N.qlen deqlen)
  then (DOWN (N.qlen-N.deqlen) / N.deqlen
        (ROUTE N).number)
2 if ((> N.use N.deuse*0.7)
    (!= 0 (NEXT N). Δ speed))
  then (UP (NEXT N). Δ speed / (NEXT N).speed N.speed)
    
```

上記のルールは、ノードの待ち行列長、ネットワークの使用率、トランザクションの応答時間と、ノードの処理速度、ネットワークの伝送容量、トランザクションのデータ量との関係をルール化したものである。DOWN, UP は修正関数である。ROUTE, NEXT は必要な識別子を検索する検索関数で、ROUTE はトランザクションが通過したノードの識別子を返す関数、NEXT は直接接続されているノードの識別子を返す関数である。

1 の例は、もし、N という識別子を持つノードの待ち行列長 N.qlen が基準値 N.deqlen より長いならば、通過しているトランザクション T のデータ量 T.number を $(T.qlen - T.deqlen) / T.deqlen$ だけ減らすことを示している。2 の例は、もし、ノード N に隣接するノード Nn の処理速度 Nn.speed を Δspeed だけ上げ、かつ、ノード N の使用率 N.use が基準値 N.deuse の 70% を超えていたら、ノード N の処理速度 N.speed を $Nn.Δspeed / Nn.speed$ 上げるといふものである。

5 おわりに

現在、以上で述べたような方式で診断システムを作成中である。本方式は、シミュレーション結果を診断するための評価基準をユーザの入力により得ている。これは、ユーザの要求に沿った性能診断を行なうためには必要な機能である。しかし、ユーザがネットワーク評価の専門家でない時、適当な評価基準を設定するのは難しいことである。また修正ルールの作成には専門家の知識が必要である。専門家の知識をうまく取り込み評価基準を設定する機能と、修正ルールのチューニング方式の開発が今後の課題である。

参考文献

[1] 宮木、一宮他：ネットワーク設計支援用のシミュレータの開発 (1), (2) 情報処理学会, 第 41 回全国大会, 1-143 ~ 1-146