社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

属性つきゴール指向要求分析法

海谷　治彦†　　佐伯　元司††　　海尻　賢二†

† 信州大学 工学部
〒 380–8553 長野市 若里 4–17–1
†† 東京工業大学 情報理工学研究科 計算工学専攻
〒 152–8552 東京都 目黒区 大岡山 2–12–1

E-mail: †{kaiya,kaijiri}@cs.shinshu-u.ac.jp, ††saeki@cs.titech.ac.jp

あらまし　本稿では，ゴール指向要求分析法 *AGORA* (Attributed Goal-Oriented Requirements Analysis Method) を提案する．AGORA では通常のゴール指向分析法の AND-OR グラフに属性値をふることで，分析者がゴールの選択肢を選別したり，ゴール間の対立を認識したり，要求変更におけるインパクトを分析したりする作業を支援する．さらに，AND-OR グラフの構造と属性値を使い，AGORA の分析結果をもとに作成される要求仕様書の品質特性を見積もることが可能となる．ここでの品質特性とは，妥当性，非曖昧性，完全性などである．この見積り値はゴールグラフ中のどのゴールを改善もしくは詳細化すべきかの指針となる．属性は優先度と貢献度の二種類があり，前者はゴールを表す節にふられ，後者は枝にふられる．優先度は stakeholder それぞれがゴールを支持する度合いを示す．貢献度はあるゴールが親ゴールの達成にどれだけ貢献するかの度合いを示す．加えて，AGORA の有効性を評価するために Web 上のユーザーアカウントシステムの要求分析を行う．

キーワード　要求分析，ゴール指向分析法，品質メトリクス

AGORA: Attributed Goal-Oriented Requirements Analysis Method

Haruhiko KAIYA†, Motoshi SAEKI††, and Kenji KAIJIRI†

† Faculty of Engineering, Shinshu University
4-17-1 Wakasato, Nagano, 380–8553 Japan
†† Department of Computer Science Graduate school of Information Science and Engineering,
Tokyo Institute of Technology
2–12–1, Oookayama, Meguro-ku, Tokyo, 152–8552, Japan

E-mail: †{kaiya,kaijiri}@cs.shinshu-u.ac.jp, ††saeki@cs.titech.ac.jp

Abstract　This paper presents the extended version of Goal-Oriented Requirements Analysis Method called AGORA, where attribute values, e.g. contribution values and preference values are added to goal graphs. An analyst attaches the contribution values and preference values to the edges and the nodes of the graph respectively during the process for refining and decomposing the goals. The contribution value stands for the degree of the contribution of a sub-goal to the achievement of its parent goal, while the preference value represents the preference of the goal for each stakeholder. These values can help an analyst to choose the alternatives of the goals, to recognize the conflicts among the goals, and to analyze the impact of requirements changes. Furthermore the values on the goal graph allow him to estimate the quality of the resulting requirements specification, such as correctness, unambiguity, completeness etc. The estimated quality values can suggest to him which goals should be improved and/or refined. In addition, we have applied AGORA to a user account system and assessed it.

Key words　Requirements Analysis, Goal Oriented Analysis, Quality Metrics

1 はじめに

ソフトウェアの要求分析過程は，大きく要求獲得と要求仕様記述の 2 つに分けることができる．要求獲得過程は，stakeholder からの情報収集を行い，顧客・ユーザ (以下，単に顧客という) の問題点やニーズ (needs) を明確にし，どのようなソフトウェアシステムを構築すればよいかの対策立案を行っていくフェーズである．このフェーズを支援する方法論には，ブレインストーミングや KJ 法などの発想法 [1], [2]，意思決定法，シナリオ分析法 [3]，ゴール指向分析法などがある．ゴール指向分析法 [4] は，顧客のニーズを，ニーズを達成するためのゴールへと徐々に分解・詳細化していき，ソフトウェアシステムに対する具体的な要求を導出していく，トップダウン形の獲得手法である．作業の結果生成されるプロダクトは，抽出されたゴールをノードとする AND-OR グラフである．文献 [5] などに事例研究を使った有効性の評価が行われており，有望な要求獲得手法の一つであろう．しかしながら，現在のゴール指向分析法は，以下の支援を行っていない．

(1) どのようにゴールを分解していけばよいかという分解の指針，

(2) 要求がコンフリクトを起こしたときの解消を行うための優先順位付け，

(3) ゴールの選択肢が複数あるとき，どのゴールを採択するかの判断基準，

(4) 要求変更があったときの影響解析，

(5) プロダクトの品質測定や測定結果をもとにした品質改善．

上記の問題を解決するために，本稿ではゴール指向分析法の AND-OR グラフに属性値を付加することによって拡張を行う．グラフのノードで表されるゴールには，stakeholder ごとの優先度 (preference) を，ゴール間の分解・詳細化関係を表す枝にはそのサブゴールが親ゴールの達成にどれだけ貢献するかの度合い，つまり貢献度 (contribution) を付加する．これらの値をもとに，ゴールの選択，コンフリクトを起こしているゴールの検出，ゴールに変更があった際の属性値を用いた影響解析が行える．

提案する手法の一つの貢献は，要求仕様書の品質計測の手法を与えたことである．ゴール指向分析法などの獲得技法や他の方法論は，プロダクト作成作業のみの支援で，プロダクトの品質を測定したり，測定結果を元に品質を改善したりする作業を支援していない．本手法では，AND-OR グラフに付加した属性値やグラフの構造的な特徴を用いて，最終的に作られる要求仕様書の妥当性，非曖昧性，完全性などの品質を予測することができ，要求獲得作業中に品質評価を行い，改善を行うことができる．これは，要求分析における品質計測をどのようにこれらの方法論に導入するかの手法を与えており，本論文では例としてゴール指向分析法に適用しているが，他の方法論にも応用できる手法である．

2 節では，例題を用いて，属性つきゴール指向分析法 (Attributed Goal-Oriented Requirements Analysis method :AGORA) を説明する．3 節では，まず IEEE 830 Standard で述べられている要求仕様書の品質項目を述べ，AGORA で作成される AND-OR グラフの属性値やグラフの構造的特徴といった因子 (quality factor) からどのように品質項目の特性が推定されるかを述べる．これらの因子の計算式も併せて形式的に定義する．最後に AGORA の適用例をもとに，利点・欠点を議論し，今後の課題に触れる．

2 AGORA の概要

2.1 AGORA の概要

通常のゴール指向分析法で支援されていない部分を補うため，AGORA でのゴールグラフは幾つかの拡張が行われている．図 1 に AGORA におけるゴールグラフのモデルを示す．

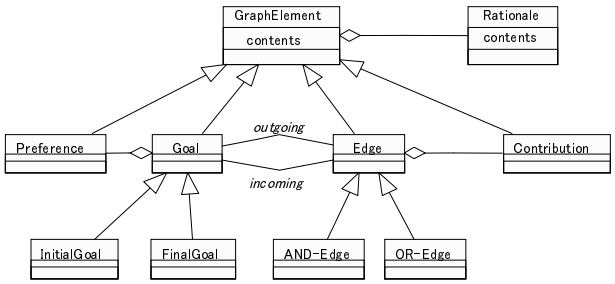

図 1 AGORA ゴールグラフのモデル

AGORA において拡張された部分は以下の通りである．

(1) 節や枝に属性値を付加する．これらの属性値とグラフ形状より 3 節で述べるように要求仕様の品質を計測することが可能になる．属性値には以下の 2 種類がある．

- 節に付加された優先度行列 ***(preference matrix)***: 各 stakeholder が節に対応するゴールを賛成もしくは満足している度合い．
- 枝に付加された貢献度 ***(contribution value)***: 枝で接続された親ゴールを達成する際にどれだけの貢献をしているかを表す度合い．

（２） 記述理由 ***(rationale)*** の付加: ゴールの選択肢を選択したり，節や枝に属性値を振たりした際の記述理由を Rationale として付加する．

AGORA に従って，属性付ゴール分解グラフを記述を作成していく手順は以下のようになる．

（１） 初期ゴールの設定

（２） ゴールの詳細化

（３） ゴールの採択

（４） コンフリクトを起こしているゴールの処理

本節の後半では「ウエブ上のユーザーアカウントシステム」を例として本手法の手順およびゴールグラフの構成を順をおって説明してゆく．

2.2 初期ゴールの設定

初期ゴールは顧客やユーザから提出されたニーズである．まず最初にこれらのニーズをゴールグラフ上の節として記述する．もし，これらのニーズに関する stakeholder の嗜好が明らかな場合には，節に対して優先度行列を振ってもよい．

ここでの例では，単に「よい Web アカウントシステムを作る」という 1 つの曖昧なゴールであるため，特に優先度行列は振らない．

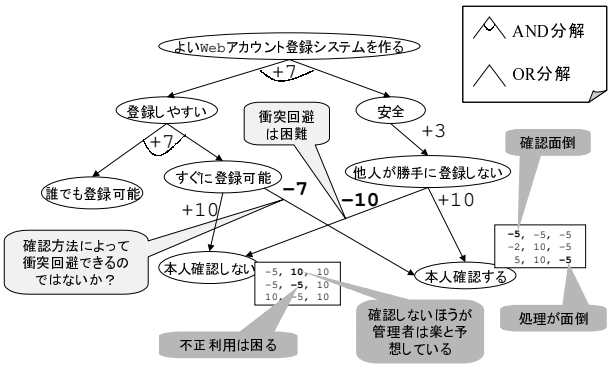

図 2　ゴールグラフ第 1 版

2.3 ゴールの詳細化

初期ゴールを起点として，ゴールをサブゴールへと分解してゆく．ここで，直接に枝によって結ばれているゴール間で，枝の起点側にあるゴールを親ゴール，終点側にあるゴールをサブゴールと呼ぶことにする．

AGORA では AND 分解と OR 分解を用いてゴールの詳細化を行う．AND 分解はサブゴールが全て達成されないと親ゴールも達成されない場合の分解であり，OR 分解はサブゴールのうちどれか一つでも達成されれば親ゴールも達成される分解である．

前述の二種類の属性値，貢献度，優先度は，それぞれゴールグラフの枝と節にふられる．また，図 2 の例にも見られるように，属性値をふった理由を枝，節ともにつけることができ，これは後に述べるゴール詳細化や選択肢の選択の支援に役立つ．

以下に属性値の値の振り方について詳説する．

- 貢献度 (Contribution Value): 親ゴールとサブゴール間をつなぐ枝にふる数値であり，−10 から 10 までの整数値をとる．この値はサブゴールが親ゴールの達成にどれだけ貢献しているかの度合いを示し，数が大きいほど貢献は大きく，その値が負の場合には，親ゴールの達成を阻害することを意味する．0 の場合，親ゴールの達成にサブゴールは関係ないことを示し，通常は枝を描かない．

OR 分解の場合は，それぞれの枝にこの値をふるが，AND 分解の場合，枝全てが揃って 1 つの意味を持つため分解に対して 1 つの値をふる．図 2 の例では，初期ゴールの分解は AND 分解で 2 つのゴールとなるが，その貢献度は，+7 である．一方，中段にある「すぐに登録可能」のゴールに対しては二つのサブゴールに OR 分解されているため，それぞれ +10 と −7 の値がふられている．

- 優先度行列 (Preference matrix): あるゴールが達成された場合，各 stakeholder はどの程度，満足するかの値を示す．この値も貢献値と同様に −10 から 10 の整数値をとる．この値は stakeholder が主観的にふってもかまわないが，AHP [6] 等を用いてもよい．

この値は単に各 stakeholder の評価を記述するだけでなく，ある stakeholder がどのように評価しているかを他の stakeholder に推測してもらい，その値も記述する．よって，単なるベクトル値ではなく行列値となる．本節の例では図 3 に示すように，顧客 (C)，管理者 (A)，開発者 (D) の三種類の stakeholder を対象としている．この行列の第 1 行は，顧客が，顧客自身，管理者，開発者の優先度を 8, −7, 0 と評価したことを表している．そして，顧客，管理者，開発者自身はそれぞれ 8, 10, 0 の優先度を持っていることが対角線を見ればわかる．

被評価者

評価者	C	A	D
C	8	−7	0
A	10	10	−10
D	5	−10	0

C = 顧客
A = 管理者
D = 開発者

図 3　優先度行列の凡例

ある stakeholder に他の stakeholder の優先度まで推測してもらう理由は，stakeholder 間の合意の度合いを測定するためである．具体的には，行列内の列

をそれぞれ数列として取り出し，その分散を計算することで，合意形成の度合いを求める．分散が 0 の場合，合意されているとみなし，分散が大きい場合，合意には遠いか，もしくは互いに誤解があることを示している．この場合，ゴールをさらに詳細化し，合意を形成するか，もしくは誤解を解消することが期待される．誤解か非合意かを判断するのに Rationale が役立つ．

図 3 の例では，管理者自身は 10 の優先度をふっているが，開発者は管理者が −10 の優先度を持つと予想しており，管理者と開発者の思惑がこの点に関して正反対であることがわかる．この場合の管理者に関する分散値は 116.3 であり，顧客に関する分散値 6.3 よりかなり大きい．この行列は図 4 中の「電子メールで折り返し確認」のゴールにふられているが，ここでの Rationale を見ると，管理者と開発者の理解が全く異なっており，ゴール分解によって，どちらかに確定する必要があることがわかる．

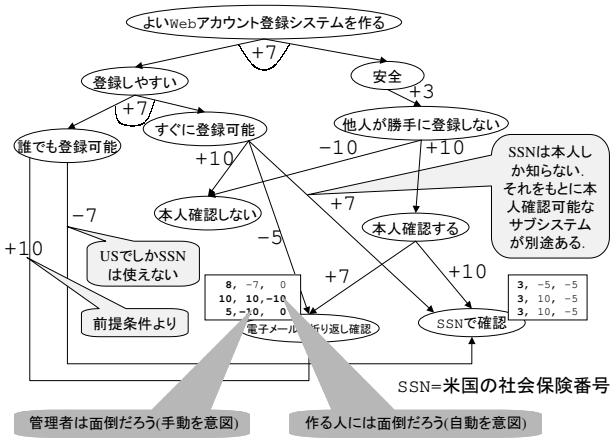

図 4　ゴールグラフ第 2 版

2.4　ゴールの採択

OR 分解で詳細化されたゴールのどれかが選択されることになるが，その判断材料として，枝に振られた貢献度や優先度行列を利用することができる．

例えば，貢献度は親ゴール達成の貢献度を示しているため，直観的に貢献度が大きな枝を採択するなどが考えられる．この実例として，図 5 での「即時自動応答&確認」のゴールと「SSN で確認」のゴールを比較を考える．「本人確認する」と「すぐに登録可能」のスーパーゴールへの貢献からは，若干，「SSN で確認」の方が貢献度が高い．しかし，「誰でも登録可能」の貢献度については，「SSN で確認」の方が −7 の負の値であるため，「即時自動応答&確認」のゴールを採択するのが妥当だと思われる．

親ゴールを共有するサブゴールを両立させなければならない場合，単純に値の大きな方を選択することはできない場合がある．例えば図 2 では，「すぐに登録可能」と「他人が勝手に登録しない」の 2 つの親ゴールは共通するサブゴールの候補をもつ．しかもこれら候補は互いに矛盾する．もし，これら親ゴールそれぞれが，単純に貢献度の大小でサブゴールの採択を行った場合，それぞれに異なるサブゴールを採択してしまい，矛盾する結果となってしまう．

2.5　ゴールのコンフリクト解消

ゴールのコンフリクトは以下の二通りの方法で検出できる．

- 貢献度が負である枝が接続されている場合
- 優先度行列の対角線の分散値が大きい場合

負の枝で接続されたサブゴールがある場合，そのサブゴールは親ゴール達成に妨げとなることを表している．負の枝を含むパスを採択せざるを得ない場合には，負の枝と接続されている親ゴールをどう扱うかを考慮する必要がある．

例えば図 2 では「すぐに登録可能」と「他人が勝手に登録しない」の二つのゴールそれぞれに負の枝を持つち，しかも互いに両立しない．ここでは，詳細化により負の枝を正に転換する可能性のある −7 の枝を採択し，図 4 に示すようにコンフリクト解決をしている．しかし，図 4 の時点でも「誰でも登録可能」と「すぐに登録可能」の 2 つのゴールを単純に達成するサブゴールは無いため，コンフリクト解消の見込みがある「電子メールで折り返し確認」のゴールを図 5 に示すようにさらに詳細化している．

採択したゴールの優先度行列において，stakeholder 自身が振った優先度 (行列の対角線) が大きく異なっている場合には，stakeholder 間にコンフリクトがあることを示している．この場合，コンフリクト解消のため，stakeholder 間で折衝を行う必要がある．例えば図 5 において左下にある 3 つのゴール (One Time Password を利用するゴール) を採択したとする．この場合，どのゴールも開発者自身の優先度は他と異なりかなり小さい．この場合，この不満を解消するため，例えば賃金や納期等の面での折衝が必要かもしれない．

2.6　ゴール詳細化の終了

ゴールの内容がソフトウェアシステムを構築するのに十分な程度に具体化された場合，ゴールの詳細化を終了する．一般に操作的な内容となっていれば，具体化したとみなしてよい．このように具体化したゴールを AGORA では最終ゴールと呼ぶ．図 5 の例では，三種類の最終ゴールのグループ (便宜のため

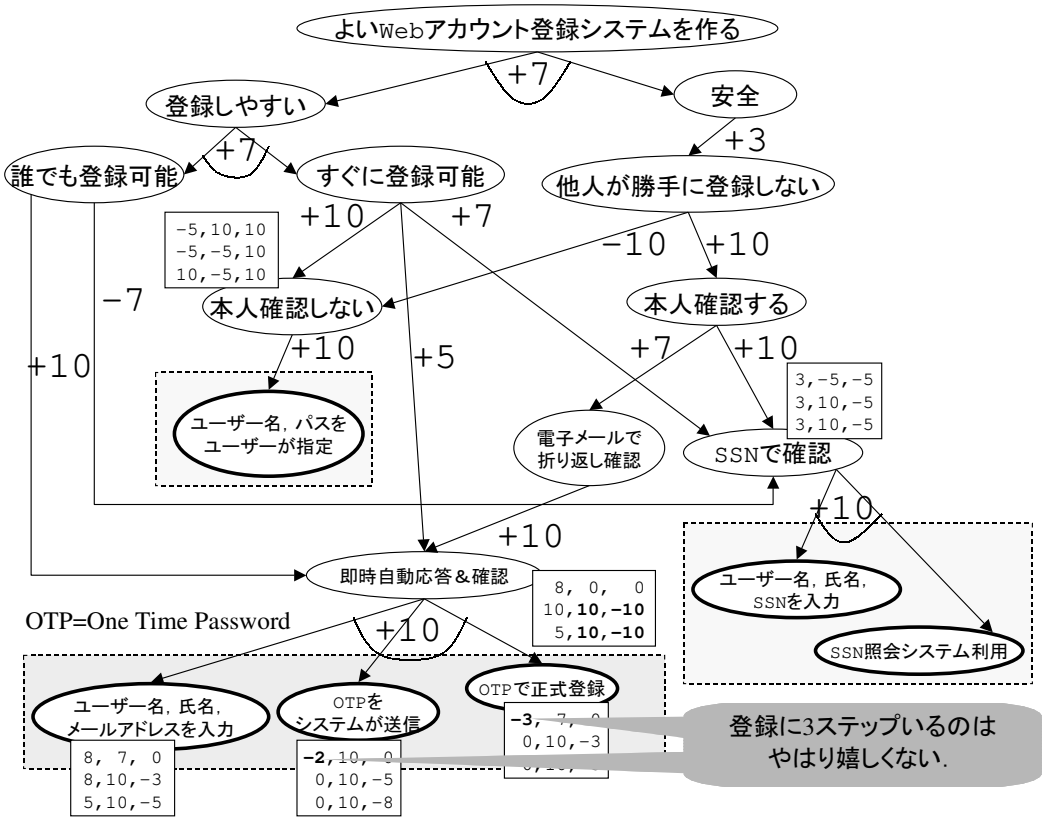

図 5　ゴールグラフ第 3 版

矩形でグループ化してある) が見られる．

3　品質の計測

3.1　要求仕様書の品質メトリックス

これまでにも McCall らによって最終製品としてのソフトウェアの品質として信頼性，効率，保守性，一貫性，使いやすさなどの品質メトリックスが定義されている [7]．これらは，ソフトウェア本体から直接計測できないため，計算精度やプログラムサイズ，標準のインタフェースやデータ型の使用度合いといった品質因子 (Quality Factor) を用いて間接的に計測している．しかしながら，これらはソースコードに対する品質メトリックス，品質因子であり，要求仕様には適用できない．

要求仕様の品質項目としては，Alan Davis [8] や IEEE Standard 830 [9] では，妥当性 (Correctness), 非曖昧性 (Unambiguity), 完全性 (Completeness), 無矛盾性 (Consistency)，検証可能性 (Verifiability)，変更可能性 (Modifiability)，追跡可能性 (Traceability)，重要性や安定性のランク付け (Ranked for Importance & Stability) を挙げている．本稿ではこれらの項目を品質メトリックスとして使用する．ただし，方法論に応じてこれらの測定可能性や重要性が異なる．例えば，形式手法については曖昧性のない形式言語で仕様が書かれるため，非曖昧性の程度を議論することは意味がないであろう．

McCall らの品質メトリックスの計測法は，比較的容易に計測可能な品質因子の値から重み係数のマトリックスを用いて品質を算出する手法を提案した．本論文でもこの手法を採用する．つまり，直接測定が容易に行える要求獲得方法論用の品質因子を用意し，それらの値をもとに，予想される要求仕様の品質メトリックスを計算する．方法論に従って作成される成果物 (作業中に作成される中間生成物も含む) から品質因子を算出するため，どのような品質因子を用いるかは採用する方法論に依存する．AGORA では，属性値つきの AND-OR グラフが生成されるため，属性値とグラフの形状を特徴とする品質因子を用いる．品質因子の定義などの詳細は次節で述べる．

3.2　品質メトリックスと品質因子

AGORA では，顧客やユーザのニーズは AND-OR グラフの初期ゴールで，要求仕様が採択された最終ゴールの集合であると考える．つまり，最終ゴールの内容の品質を考えることにより，要求仕様記述過程で最終ゴールの内容からできあがってくる要求仕様書の品質を予測することができる．また，その品質評価結果によって，どのゴールを改善したり，詳細化しなおしたりしなければならないかを，要求獲

得過程 AGORA の作業中に知ることができる．つまり，仕様書が完成する前にその品質改善作業を行うことができる．

本稿は，AGORA で計測可能な品質メトリックスや品質因子をすべて列挙することを目的としているのではない．これらの一部と，それを使ってどのように品質メトリックスが計算されるかを例示することによって，AGORA のメトリックス計測可能性を示すことが本稿の目的である．

本稿で例示する品質因子の形式的な定義は，図 1 の AND-OR グラフの定義を用いて行っている．図中のクラス名はそのまま集合を，関連名は述語を表しているものとする．例えば，$FinalGoal$ は最終ゴールの集合を表し，$f \in FinalGoal$ で f は最終ゴールの 1 つを表している．ゴール g が優先度 m を持っていることを $has(g,m)$ で表す．has は集約を表す述語名として使用している．$incoming$ のように関連名が指定されているときは，それをそのまま使用し，$incoming(g,e)$ でゴール g に枝 e が入ってきていることを表すものとする．

妥　当　性

要求仕様書中に書かれている要求はすべて，顧客のニーズを満たしていることを表すメトリックスである．従って，AGORA のグラフ上で採択された最終ゴールに至るパス上の貢献度がどうなっているかが妥当性のメトリックスに大きな関連がある．それ以外にも最終ゴールに付加された顧客の優先度が高ければそのゴールは妥当性が高いといえるであろう．

ここでは，妥当性に関連した品質因子として，

- 最終ゴールに至るパス上の貢献度の最小値の平均値 (Sat)，
- 貢献度が正の枝の割合 (Pos)，
- 最終ゴールの顧客の優先度の平均値 (Cup)

を用いることにする．これらの値は，図 1 の AND-OR グラフのモデルを用いると以下のように定義できる．

$$Sat = (AVE_{f \in FinalGoal}\{Sat(f)\})/10$$
$$Sat(f) = AVE_{g \in \{g \in InitialGoal | \exists p \cdot p \in path(g,f)\}}\{Ctrb(f,g)\}$$
$$Ctrb(f,g) = MIN_{\{e \in path(g,f)\}}\{c \in Contribution | has(e,c)\}$$

$$Pos = \frac{\#\cup_{g \in IntialGoal, f \in FinalGoal} PositivePath(g,f)}{\#\{p \in path(g,f) | g \in InitialGoal \wedge f \in FinalGoal\}}$$
$$PositivePath(g,f) = \{p \in path(g,f) | \forall e \in p, \forall c \in Contribution \cdot (has(e,c) \rightarrow (c > 0))\}$$

$$Cup = AVE(\cup_{f \in FinalGoal, s \in Stakeholder, m \in Preference}\{m_{s,customer} | has(f,m)\})$$

ただし，$AVE_{p(x)}\{s(x)\}$，$MIN_{p(x)}\{s(x)\}$ は各々，述語 $p(x)$ を満足する x から構成される数値集合 $s(x)$ の平均値，最小値を求める関数である．$path(g,f)$ はゴール g から f に至るパス (枝の列) の集合を表す．$m_{s,customer}$ は，ゴール (グラフのノード) につけられた優先度マトリックス m 中の，Stakeholder s が採点した顧客の優先度を表す．

図 5 で，採択された最終ゴールに至る 3 つのパスについて，Sat を計算すると，$(10+5+7)/3/10 = 0.73$，Pos は 3/3=1.0，Cup は $((8+8+5)+(-2+0+0)+(-3+0+0))/(3+3+3)/10$=0.18 となる．

非曖昧性 **(Unambiguity)**

仕様が読者にとって 2 つ以上の解釈を持たないことである．AGORA では，ゴールの記述内容の非曖昧性が考えられる．内容は自然言語で記述されるため，その表現が曖昧な言葉をどれだけ含んでいないかで計測できる．自然言語処理技術が向上すれば意味まで踏み込んだ計測が可能になるかもしれない．問題領域が決まっていれば，Stakeholder の誰でも一意に解釈可能な領域固有の共通な語彙をあらかじめ決めておき，それらの含有率によって非曖昧性を推定できるであろう．

これ以外には，ゴールの内容ではなく，優先度マトリックスの値を使って推測することも可能である．Stakeholder が採点した優先度で，ある特定の Stakeholder に対する値の分散，つまり図 5 のマトリックスの縦方向の分散が大きければ，Stakeholder 間での共通認識ができていないことが予想され，その原因としてゴールの記述内容が曖昧であることが予想される．この因子は，以下のように定義される．

$$Vdv = 1 - \frac{AVE_{s1 \in Stakeholder, f \in FinalGoal}\{Vd(s1,m)) | has(f,m)\}}{10}$$
$$Vd(s,m) = AveDev\{\cup_{i \in Stakeholder} m_{i,s}\}$$

ここで，$AveDev(S)$ は数値集合 S の平均値に対する各要素の絶対偏差の平均値を表す．

図 5 では，Vdv=$1-(1.33+1.33+1.78+0.89+0+2.89+1.33+1.33+1.78)/9/10) = 0.14$ となる．この例では，全最終ゴールの平均をとっているが，個々のゴールの値をみることにより，どのゴールが曖昧であるかの指針に使用することもできる．例えば「OTP をシステムが送信」というゴールにおいて，開発者に対する優先度の絶対偏差の平均は 2.89 ともっとも高いため，このゴールの開発者に対する Stakeholder

間に認識のずれが大きいことを表しており，曖昧な可能性がある．

完全性 (**Completeness**)

一般的には記述漏れがないことを指すが，ここでは顧客のニーズをすべて仕様化したかどうかで評価する．

AGORA では，初期ゴールが顧客のニーズを表しているため，初期ゴールがすべて正の値の貢献度を持つ枝のみで最終ゴールへ展開されたかどうかという網羅性が完全性のメトリックスに大きな関連がある．この因子は以下のように定義できる．

$$Cov = \frac{\#\{i \in InitialGoal | \exists f \in FinalGoal \cdot AllPositive(i,f)\}}{\#InitialGoal}$$

$$AllPositive(i,f) = (PositivePath(i,f) = path(i,f))$$

図 5 の場合は，初期ゴールはいずれも採択された最終ゴールへと，すべて貢献度が正の枝で接続されているため，$Cov = 1$ となる．この他にも，非曖昧性のところで述べた，曖昧語が含まれていない割合，領域固有の語の含有割合なども，未定義語や未掲出語がどれだけ含まれていないかに関連があるため，完全性のメトリックスと関連のある因子となりうる．

無矛盾性 (**Consistency**)

記述内容が意味的にお互いに矛盾がないということを示すメトリックスである．

これに関係のある因子の例としては，妥当性にときに挙げた，採択したパスのなかで貢献度が正の枝のみを持つパスの割合 Pos がある．負の枝を持つゴールが採択されていた場合，そのゴールは接続されているゴールの達成を妨げようとしているため，相反するゴールがともに選ばれていることになり矛盾である可能性がある．その他に，優先度マトリックスの Stakeholder 間の値の分散も無矛盾性に関連がある．つまり図 5 のマトリックスの横方向の分散が大きければ，Stakeholder 間で優先度が競合を起こしており，そのゴールが Stakeholder 間で矛盾を引き起こしている可能性がある．この因子は，以下のように定義される．

$$Hdv = 1 - \frac{AVE_{s1 \in Stakeholder, f \in FinalGoal}\{Hd(s1,m))|has(f,m)\}}{10}$$

$$Hd(s,m) = AveDev\{\cup_{i \in Stakeholder} m_{s,i}\}$$

図 5 では，$1 - (3.33 + 5.33 + 5.56 + 4.89 + 5.56 + 6.22 + 3.78 + 5.11 + 5.56)/9/10 = 0.5$ となる．

検証可能性 (**Verifieability**)

AGORA では，記述本体が自然言語という非形式言語で記述されているため，テストによる検査しかない．最終ゴールの記述からテストケースが生成しやすいかどうかが 1 つの因子になり得るであろう．そのためには，最終ゴールの記述内容に曖昧語が含まれていない割合，記述が操作的になっている割合 (主動詞が動作を表す動詞である割合) などが因子として挙げられる．

変更可能性 (**Modifierability**)

採択したパス中に含まれるノード 1 つあたりに入ってくる枝の数が多い場合，そのノードが達成を支援しているゴールの数が多いことになり，それを変更すると，対処しなければいけないゴールの数が多いことになる．この因子は，AND-OR グラフがどれだけ木に近いかを表す尺度であり，

$$Tre = \frac{\#\{g \in RefinedGoals | \#\{e | incoming(g,e)\} = 1\}\}}{\#RefinedGoals\}}$$

$$RefinedGoals = Goal - InitialGoal$$

と定義できる．図 5 では，$(11 - 2)/11 = 0.82$ となる．

追跡可能性 (**Traceability**)

追跡可能性は，AGORA 内でのもの (intra-traceability) と，次の段階 (要求仕様記述段階) に対するもの (inter-traceability) との 2 種類ある．後者はゴール指向分析法で得られた成果物を次の要求仕様記述過程でどのように使用するかに依存している．そのため，記述方法論に何を使用し，それとどのように連結するかが判明しないと測定できない．AGORA 内だけの追跡可能性を計測するための因子としては，最終ゴールが顧客のニーズから分解されてきたかどうかの連結度 Con や Rationale がどれだけ付けられているかの度合い Rat が考えられる．

$$Con = \frac{\#\{f \in FinalGoal | \exists g \in InitialGoal, \exists p \cdot p \in path(g,f)\}}{\#FinalGoal}$$

$$Rat = \frac{\#\{g \in GraphElement | \exists e \in Rationale \cdot has(g,r)\}}{\#GraphElement}$$

例では，採択された 3 つの最終ゴールはすべて初期ゴールを分解したものであるため，$Con = 1$ である．

重要性や安定性のランク付け (**Ranked for Imporatnce & Stability**)

これは必要な個所に要求の重要性や安定性の順位を明記しているかどうかの品質メトリックスである．AGORA では，各ゴールに優先度マトリックスを付加できるようになっているため，重要性の指定は間接的に可能ではある．しかし実際にそれらを本当に記述しなければならないかの判断はできないため，関連の深い因子は定義が難しい．優先度マトリックスがつけられているゴールの割合が 1 つの指針にはなる．

表 1　品質マトリックスの例

	Sat	Pos	Cup	Vdv	Cov	Hdv	Tre	Con	Rat
妥当性	0.5	0.3	0.2						
非曖昧				1.0					
完全性					1.0				
無矛盾		0.6			0.4				
変更可							1.0		
追跡可								0.7	0.3

3.3　品質メトリックスの計算

各品質メトリックスは，前節で述べた品質因子を加重平均することによって求められる．つまり，品質メトリックス $QM_i(i = 1,\ldots,n)$ は，品質因子の値 $QF_j(j = 1,\ldots,m)$ と重み $w_{i,j}(i = 1,\ldots,n, j = 1,\ldots,m)$ を用いて以下のように計算される．

$QM_i = \Sigma_{j=1,m} w_{i,j} \times QF_j$

ただし，$\Sigma_{j=1,m} w_{i,j} = 1$ である．

重みづけのマトリックス $(w_{i,j})(i = 1,\ldots,n,\ j = 1,\ldots,m)$ を品質マトリックスと呼び，これは経験にもとづき方法論ごとに事前に与えられるものとする．

AGORA の場合は，一例として表 1 が挙げられる．表中で空欄は 0 を表す．これは本稿で形式的に定義した品質因子に限ったものであり，説明用の一つの例であることに留意されたい．その意味で検証可能性と重要度や安定性のランク付けははずしてある．前節の例を用いて妥当性度を計算すると，

$0.5 \times Sat + 0.3 \times Pos + 0.2 \times Cup =$
$0.5 \times 0.73 + 0.3 \times 1.0 + 0.2 \times 0.18 = 0.7$ となる．

4　おわりに

本稿では，ゴール指向の要求分析法の AND-OR グラフに優先度と貢献度という属性値をふることで，要求分析を支援するための分析法 AGORA を提案した．さらに，AND-OR グラフをもとにして記述される要求仕様書の品質特性を，グラフの構造と属性値の値をもとに見積もる手法を提案した．

属性値自体を決定する過程の支援は本手法では対象外であるが，この値を適切にふることは本手法を支える重要な点である．例題記述の際には，特に優先度行列を決定するのは困難であった．これら属性の決定においては前述のように AHP 等の手法や，インタビュー支援技法等との連携が有功だと思われる．

図 5 の例では，最終ゴールを操作的な記述まで詳細化した．これによって，例えば最終ゴールをユースケースとみなし，AGORA のゴールグラフをもとにしてユースケース図を記述する工程と連結することが可能なように思われる．AGORA を後段の工程とどのように連結するか，どのような記述方法論と連結するかについては今後の課題であるが，シナリオ記述やユースケース記述は AGORA と親和性の高い有望な手法であると思われる．また，記述方法論との連結の議論を通して，追跡可能性に関する品質項目の計測をさらに詳しく行うことが期待できる．

また，これは AND-OR グラフの特性であるが，サブゴールが複数の親ゴールを持つことで，操作的なゴール記述の漏れを防ぐ効果があるように思われる．例えは図 5 の「即時自動応答&確認」を，「本人確認する」という観点のみから見た場合，「即時自動応答&確認」の操作的なサブゴールのうち最初の処理にあたる「ユーザー名，氏名，メールアドレスの入力」の記述を落としてしまった．この記述漏れは，「すぐに登録可能」などの “登録する” という観点からのスーパーゴールとの関係によって防ぐことができた．

文　献

[1] D. Bellin and S. Simone. *The CRC Card Book*. Addison Wesley, 1997.

[2] 川喜田二郎. 発想法. 中公新書, 1967.

[3] 郷健太郎. ユーザの視点を取り入れる技術：システム開発におけるシナリオの役割. 情報処理, Vol. 41, No. 1, pp. 82–87, 2000.

[4] A. van Lamsweerde and E. Letier. Integrating Obstacles in Goal-Driven Requirements Engineering. In *Proc. of 20th International Conference on Software Engineering*, pp. 53–63, 1998.

[5] A. Anton and C. Potts. The use of goals to surface requirements for evolving systems. In *Proc. of 20th International Conference on Software Engineering*, pp. 157–166, 1998.

[6] 刀根薫. ゲーム感覚意志決定法 AHP 入門. 日科技連, Mar. 1986.

[7] Joseph P. Cavano and James A. McCall. A Framework for the Measurement of Software Quality. In *Proc. of ACM Software Quality Assurance Workshop*, pp. 133–139, Nov. 1978.

[8] Alan M. Davis. *Software Requirements Objects, Functions, and States*. Prentice Hall, 1993.

[9] IEEE Recommended Practice for Software Requirements Specification, Oct. 1998. IEEE Std 830-1998, ISBN 0-7381-0332-2 SH94654(Print).