



(51) International Patent Classification:
G06N 5/00 (2006.01)

(21) International Application Number:
PCT/JP2012/005351

(22) International Filing Date:
27 August 2012 (27.08.2012)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant (for all designated States except US): **NEC CORPORATION** [JP/JP]; 7-1, Shiba 5-chome, Minato-ku, Tokyo, 1088001 (JP).

(72) Inventor; and

(75) Inventor/Applicant (for US only): **TARASENKO Sergey** [RU/JP]; c/o NEC CORPORATION, 7-1, Shiba 5-chome, Minato-ku, Tokyo, 1088001 (JP).

(74) Agent: **UDAKA, Katsuki**; No2, Azuma Bldg. 5fl, 14, Kandasakumacho 1-chome, Chiyoda-ku, Tokyo, 1010025 (JP).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM,

AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

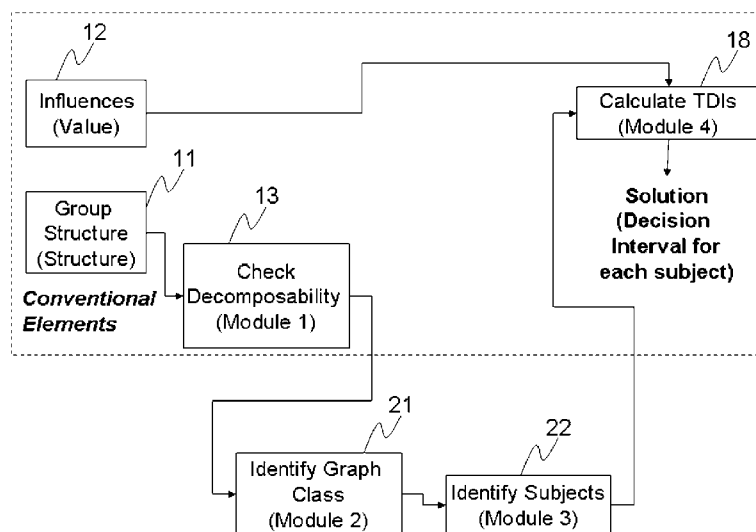
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: METHOD AND SYSTEM FOR PERFORMING REFLEXIVE GAME THEORY (RGT) INFERENCE

[Fig. 9]



(57) Abstract: The present invention is a method of performing Reflexive Game Theory (RGT) inference comprising: pre-computing of possible group structure and real-time selecting (template matching) of the group templates by using the input information.

Description

Title of Invention: METHOD AND SYSTEM FOR PERFORMING REFLEXIVE GAME THEORY (RGT) INFERENCE

Technical Field

[0001] The present invention relates to a method and a system for performing Reflexive Game Theory inference.

Background Art

[0002] Reflexive Game Theory (RGT) enables prediction of group members (subjects) decisions. This analysis is based on only relationships (conflict or alliance) between subjects (individuals, teams or work groups, companies, etc.) and their mutual influences Non-Patent Documents 1 and 2. Conventional inference algorithm and example of RGT inference are presented in Figs. 1 and 2, respectively. The groups of subjects are presented in the form of fully connected graphs. The graph can have dashed-line ribs (conflict relationships) and solid-line ribs (alliance relationships).

[0003] The stages of RGT analysis (inference) based on symbolic computations are:

- 1) input group structure as pair-wise relations between subjects (11);
- 2) input subjects mutual influences (12);
- 3) check graphs decomposability (13);
- 4) construct a polynomial (14);
- 5) perform Diagonal Form Transformation(DFT)(15);
- 6) build decision equations (16);
- 7) perform transformation of the decision equation into canonical form (TDECF) to solve decision equation, obtain templates of decision intervals for each subject (17);
- 8) input mutual influences into the templates to get possible decisions (18).

Citation List

Non Patent Literature

[0004] NPL 1: Lefebvre, V.A. Lectures on Reflexive Game Theory, Cogito-Centre, Moscow, 2009.
NPL 2: V.A. Lectures on Reflexive Game Theory, Leaf & Oaks Publishers, 2010.

Summary of Invention

Technical Problem

[0005] During each RGT inference the DFT and TDECF should be performed. Both DFT and TDECF are iterative procedures, which involve non-trivial operations (Fig.2). On the other hand, for the group of three subjects, there are only 8 different graphs (group structures). This is illustrated in the left part of Fig. 3. Therefore, repetitions of both

transformations create redundancy of calculations and increase overall computation time.

[0006] This problem can be solved by pre-computing templates of decision results for each group and just input actual parameters into the templates each time. However, the number of different graphs (group structures) grows exponentially as $2^{n(n-1)/2}$, where n is a number of group members (subjects). Thus for groups of four and five subjects there are $64(=2^6)$ and $1024(=2^{10})$ different graphs, respectively. I call this straight-forward pre-computation. Therefore, the straight-forward pre-computation results in combinatorial burst and drastic increase in storage size.

[0007] Since I have to store a template of decision interval for each subject, the total memory (storage) size required to store pre-computed template is $\text{MemSize} = n2^{n(n-1)/2}$. Therefore, there are two problems: 1) computational redundancy when using standard approach; and 2) combinatorial burst causing drastic increase in storage size when using straight-forward pre-computation.

Solution to Problem

[0008] The present invention is a method of performing Reflexive Game Theory (RGT) inference comprising: pre- computing of possible group structure and real-time selecting (template matching) of the group templates by using the input information.

[0009] The present invention is a method of processing the actual input data for RGT inference by using pre-computed templates comprising: identifying a certain template corresponding to the input data by unique structure features; and identifying location of each subject by means of position feature.

[0010] The present invention is a system of performing Reflexive Game Theory (RGT) inference comprising: pre- computing module that pre-computes possible group structure, and real-time selecting module that real-time selects (template matching) the group templates by using the input information.

[0011] The present invention is a system of processing the actual input data for RGT inference by using pre-computed templates comprising: module that identifies a certain template corresponding to the input data by unique structure features; and module that identifies location of each subject by means of position feature.

Advantageous Effects of Invention

[0012] The present invention makes it possible to avoid redundancy of computation and reduce the memory size required.

Brief Description of Drawings

[0013] [fig.1]Figure 1 is a block diagram showing the functional structure of the conventional RGT inference.

[fig.2]Figure 2 is explanatory diagram showing example of conventional RGT

inference.

[fig.3]Figure 3 is explanatory diagram showing example of generalization procedure.

[fig.4]Figure 4 illustrates examples of non-decomposable graphs.

[fig.5]Figure 5 illustrates examples of sub-classes for class Cl ($k = 2$; $n = 4$).

[fig.6]Figure 6 illustrates example of generalization procedure for the group of four members, indicating classes, sub-classes and graph types.

[fig.7]Figure 7 illustrates example of how to make Templates of Decision Intervals.

[fig.8]Figure 8 is a flow chart for Module 1.

[fig.9]Figure 9 is a block diagram showing the general schema of invention.

[fig.10]Figure 10 is a block diagram showing the general schema of invention to be realized by means of hierarchical LUT.

[fig.11]Figure 11 is an explanatory diagram showing example of hierarchical LUT.

[fig.12]Figure 12 is a block diagram showing the exemplar schema of invention to be realized by means of hierarchical LUT.

[fig.13]Figure 13 is a block diagram showing example of data flow when invention based on hierarchical LUT is used for RGT inference.

Description of Embodiments

[0014] The invention aims to solve both problems of computational redundancy and exponential growth by providing a trade-off between number of operations and storage size

[0015] To avoid the computational redundancy and achieve smaller size of the storage, pre-computation method different from the straight-forward computation is used.

[0016] The present method uses classification of the graphs for the given number of subject. For examples, in the case of three subjects, there can be 8 different graphs. These 8 graphs can be generalized to extract four different graph classes regarding the number of alliance and conflict relationships (rib types) in a group. The original 8 graphs are shown in the left part and the graph templates (classes) are shown in the right part of the Fig.3.

[0017] In the case of four group members, the generalization procedure of the graph is slightly more complex. First, for groups of four members, there can be non-decomposable graphs, which cannot be processed by means of RGT. The non-decomposable graph $S_{(4)}$ for the group of four members is presented in the left part of Fig.4.

[0018] Second, there can be sub-classes in the given class. The example of sub-classes for the class with two alliance relationships is shown in Fig. 5.

[0019] In Fig. 6, the result of generalization procedure in the case of a group of four members is presented. The general notification of the class is Cl($k;n$), where k is a

number of alliances (solid lines), and n is a number of group members.

[0020] The left part of Fig.6 represents examples of graphs of each type, the center part indicate the graph Templates and associated 4-dim vectors (see below for description). The right part illustrates distinction between classes, sub-classes and types. The important difference between actual graphs and templates is that in the former one actual subject variables are assigned to the graph vertices, while template contains no information about actual subject variable. Therefore, I have to relate the vertices of the actual input graph to the ones of the corresponding template.

[0021] Next I present the general structure based on the proposed idea of graph classification.

[0022] Each group of subjects (graph) is characterized by number of subjects and the relationships between them. Each subject is represented as a single vertex of a graph. The dashed ribs represent conflict relationships, and solid ribs represent the alliance relationships.

[0023] I consider n -dim vector (Fig.6), where n is a number of group members, which shows number of solid ribs connected to the given vertex. Therefore, such vector is an algebraic representation of a group (graph). For example, let $(a_s, b_s, c_s, \dots)$, x_s - is a number of solid lines for subject x . In other words, x_s is a Number of Alliances (NoA) for subject x .

[0024] This vector is directly related to the structure of the actual graph. In reality, a set of 4-dim vector correspond to a particular graph type. Therefore, I need a feature which can provide intra-class invariance and inter-class separability: the value of this feature for different graphs of the same type is the same, but feature values corresponding to the different classes are different.

[0025] Using such vector I calculate the feature values to uniquely identify each template. The feature is a 2-dim vector:

$$\text{Feat} = (\text{Sum} + \text{Max} + \text{Min} + \text{cntMin}; \text{Sum} + \text{Max} + \text{Min} + \text{cntMax}) \quad (1)$$

where $\text{Sum} = a_s + b_s + c_s + \dots$, $\text{Max} = \max\{a_s; b_s; c_s; \dots\}$, $\text{Min} = \min\{a_s; b_s; c_s; \dots\}$, and cntMin and cntMax represent the number of Min and Max values, respectively, in the vector.

[0026] Each template has n corresponding decision intervals. Since the decision intervals are no longer linked to particular subjects, the decision intervals are formulated in terms of abstract subject variables subj1 , subj2 , etc. I call these intervals to be templates of decision intervals (TDIs).

[0027] For example, for group $ab+c$ the decision intervals are $b+c \supseteq a \supseteq c$, $a+c \supseteq b \supseteq c$ and $1 \supseteq c \supseteq ab$. The corresponding TDIs are $\text{subj1} + \text{subj2} \supseteq \text{subj3} \supseteq \text{subj2}$, $\text{subj3} + \text{subj2} \supseteq \text{subj3} \supseteq \text{subj2}$ and $1 \supseteq \text{subj2} \supseteq \text{subj1} \text{ subj3}$ (Fig. 7).

- [0028] To relate the vertices of the input graph to the selected template I use second feature for each vertex. This feature (Feat1) is NoA of the given vertex (subject). The general structure of the present invention based on the original pre-computation method includes the following modules: 1) module 1 - check graph decomposability; 2) module 2 - identify the graph type (template), which fits the given group structure; 3) module 3 - relate vertices of the input graph to the one of the template's (bind real subjects and abstract subjective variables); 4) module 4 - calculate values for decision intervals.
- [0029] Module 1. For three subjects, there can be no non-decomposable groups. In the case of four subjects, there is one non-decomposable graph ($S_{(4)}$)(Fig.4, left part). The feature that uniquely identifies $S_{(4)}$ is Feat = (11; 11). When there are more than four subjects ($n > 4$), all the non-decomposable graphs contain the sub-graph isomorphic to the $S_{(4)}$. Therefore, I have to consider all possible sub-graphs, consisting of four subjects, and calculate Feat value for each of them. If there are no non-decomposable sub-graph, Feat value (11;11) should not appear. The flow chart for Module 1 is presented in Fig. 8.
- [0030] Module 2. If the group is decomposable, it is possible to uniquely identify the template for the graph by using Feat value. Each template has associated Feat value. The Feat values for the graph and the stored templates are compared element-wise.
- [0031] Module 3. Each template of a graph has TDIs associated with it. To bind the actual subjects (a, b, c, etc.) to the abstract subject variables (subj1, subj2, subj3, etc.), the vector elements x_s , x is either of a, b, c, etc., are used as features. Here I consider sample case. Let the group contains four subjects a, b, c and d, and the corresponding vector is (1,2,0,1). That means that subject b is in alliance with subjects a and d, who are in conflict with each other. Subject c is in conflict with other subjects. This group is described by polynomial $b(a+d)+c$.
- [0032] Each abstract subject variable has associated feature value. The feature (Feat1) is number of connected solid ribs (x_s). Let, Feat1(subj1)=2, Feat1(subj2)=0, Feat1(subj3)=1 and Feat1(subj4)=1. Using Feat1, it is possible uniquely identify that subj1 is subject b, subj2 is subject c.
- [0033] On the other hand, both subj3 and subj4, and a and d, have the same associated value 1. But, in this case the decision intervals for subject subj3 and subj4 are symmetric:
- $$subj1 + subj2 \text{ not}(subj1) \supseteq subj3 \supseteq subj1 + subj2 \text{ not}(subj1) \quad (2)$$
- $$subj1 + subj2 \text{ not}(subj1) \supseteq subj4 \supseteq subj1 + subj2 \text{ not}(subj1) \quad (3)$$
- Therefore, subjects a and d can be arbitrarily associated with the abstract subjects subj3 and subj4.
- [0034] Module 4. Once the actual subjects are bound to the abstract ones, it is possible to

calculate the values for limits of the templates of decision intervals (TDIs), by inputting the influences from the influence matrix. A user should fill in the influence matrix before implementing the RGT inference. The values from influence matrix are input into the TDIs corresponding to the template.

[0035] The general schema of invention is presented in Fig. 9.

[0036] The present invention is based on the template matching approach: 1) I use a special feature, which can be calculated for each template; and 2) unique values are associated with each template.

[0037] The exploitation of the present invention implies two phases.

[0038] During the off-line (preparation) phase, the required templates and corresponding feature values are stored in the memory units system or device.

[0039] When RGT inference is performed (on-line phase), the values of features are calculated on the basis of input data in real-time and compared with the feature values associated with each template in memory.

[0040] During the off-line phase, the templates (graph classes and sub-classes) are obtained by generalization of graphs using methods of combinatorics.

[0041] The total number of graphs of the given class $Cl(k; n)$ is $C_{n(n-1)/2}^k$, where

$$C_{n(n-1)/2}^k = \frac{(n(n-1)/2)!}{(n(n-1)/2 - k)!k!}$$

is a number of all permutations for k out of $n(n-1)/2$.

[0042] The total number of classes (TNC) is given by formula:

$$\text{TNC} = n(n-1)/2 + 1 \quad (4)$$

For $k > 1$, there are possible sub-classes regarding mutual location of the ribs in a graph. For example, for $k = 2$, there are two sub-classes (Fig.5).

[0043] The graphs belonging to different sub-classes are not isomorphic. The graphs belonging to different classes are also not isomorphic. Consequently, graphs belonging to the sub-classes of different classes are also not isomorphic.

[0044] Therefore, I consider graph types - each sub-class of any class is considered to be a distinct graph type.

[0045] Finally, there is a big class of graphs, which cannot be processed in RGT. These graphs are called non-decomposable graphs (Fig. 4). The non-decomposable graphs can be found as sub-class of the classes,

when $k \geq 3$.

[0046] In the case of $k = 3$ and $n = 4$, there is a single non-decomposable graph called $S_{(4)}$ (Fig. 4).

[0047] I consider the non-decomposable graphs as a separate independent class (type). Therefore, formula (4) is updated to:

$$TNC = n(n-1)/2 + 2 \quad (5)$$

[0048] The overall total number of types (TNT) is a sum of sub-classes for each class plus one (a type for non-decomposable graphs):

$$TNT(n) = \sum_{k=0}^{n(n-1)/2} TNS(k) + H(n-4) \quad (6)$$

where $TNS(k)$ is a total number of sub-classes for the given class $Cl(k; n)$, excluding sub-classes containing non-decomposable graphs; $H(\cdot)$ is Heavyside step function, which produces output 1, if $n \geq 4$, and 0 otherwise. $TNS(k)$ can be calculated by means of combinatorial methods in each particular case.

[0049] The overall memory size (MemSize) required to store all graph types and their corresponding TDIs for the given n is

$$MemSize(n) = n \left(\sum_{k=0}^{n(n-1)/2} TNS(k) \right) \quad (7)$$

[0050] The present invention allows to avoid redundancy of computation and reduce the memory size required.

[0051] The present invention allows to sufficiently reduce the size of the storage required. For example, $TNT(3)$ is 4, $TNT(4)$ is 11, and $TNT(5)$ is 32, while corresponding numbers for straight-forward pre-computing are 8, 64 and 1024, respectively.

[0052] The corresponding $MemSize(n)$ values to store all required graphs types and corresponding TDIs for groups of three ($n=3$), four ($n=4$) and five ($n=5$) subjects are 12, 40, and 155, respectively.

[0053] As a method to carry out the invention, I suggest to use the combination of LookUp Tables (LUTs). The general schema of invention realized by means of LUTs is shown in Fig. 10. The term Hierarchical LUT in Fig. 10 means that several LUTs are connected consecutively. Word hierarchical implies that each LUT represent a certain level of abstraction of information.

[0054] The general schema of a hierarchical LUT for RGT inference is presented in Fig. 11.

[0055] The detailed schema of the invention realized by means of LUTs is presented in Fig. 12. First, user inputs number of subjects (NS) (41), subject variables (42) and the group structure by indicating pair-wise relations (43) between subjects. Relations can take value either 0 or 1 meaning conflict or alliance, respectively. The relations are

formalized in the form of pairs of subject variables: for example, $ab=1$, meaning subjects a and b are in alliance.

- [0056] The pairs of subjects are then propagated into module (45) to check, whether the input group structure is decomposable. If the answer is " yes ", data about the group structure are propagated into module (46). The process of RGT inference is stopped providing no solution, otherwise.
- [0057] If input graph is decomposable, relationship pairs (ab, bc, ad, etc.) are input into the module (46) from module (43). Using relationship pairs, 4-dim is created and Feat value is calculated in module (46). Module (46) outputs Feat value.
- [0058] The modules (48), (49) and (412) are storages. Module (48) contains one LUT, which contains distinct numbers of subjects, associated with the Feat values for graph types (Fig. 11). The LUTs of Feat value for graph types are stored in module (49). In storage (48) the numbers of subjects are associates with particular LUTs in module (49). Therefore output of module (48) is the reference to a particular LUT in module (49).
- [0059] In module (49), each LUT contains the Feat values associated with a particular set of TDIs. The TDIs for each Feat value are stored in module (412) in separate LUTs. Each LUT in module (412) contains NoS for each abstract subject and corresponding TDI. Therefore, module (49) outputs the reference a particular LUT in module (412).
- [0060] Module (47) takes two inputs from the module (41) and module (48). Module (47) outputs the reference to a certain LUT in module (49). This reference is passed to module (410).
- [0061] Module (410) takes three inputs from modules (47), (46) and (49). The input from module (47) is a reference to a particular LUT in module (49). Having selected a particular LUT in module (49), module (410) uses input from module (46) to match a particular record in the LUT from module (49). Module (410) outputs the reference to a particular LUT in module (412).
- [0062] The module (410) receives inputs from modules (46) and (49). Module (46) provides Feat value, while module (49) is a LUT of group templates. Module (410) performs selection a group configuration by comparing Feat value from (46) with the ones in the LUT (49). Module (410) outputs LUT (412).
- [0063] Module (411) receives inputs from modules (43) and (45) and outputs feature (NoA) value for each subject variable (a, b, c, ...) to module (413). The module (413) receives three inputs from modules (410), (411) and (412). the input from module (410) is used to select a particular LUT in module (412). The input from module (411) is used to related the actual subjects variables to abstract subject variables. Module (413) outputs the collection of TDIs, which are bind to particular actual subjects.
- [0064] Finally, module (414) takes inputs from Influence module (44) and module (413).

Module (414) calculates values of limits of TDIs, using the information about mutual influences from module (44). Module (414) outputs solution of RGT inference.

Example 1

[0065] <Example of implementation in the case of a group of four members.>

[0066] Here the example of implementation of the invention is presented. The example is presented in Fig.13.

[0067] User inputs number of subjects NS=4 in Module (41).

[0068] User inputs subject variables a,b,c,d in Module (42).

[0069] User inputs pair-wise relationships ab=0, ac=0, ad=0, bc=0, bd=1, cd=0 in Module (43).

[0070] User inputs influences in Module (44). The Influence matrix is presented in Fig.13 (44).

[0071] Module (45) performs checking of decomposability of the input graph. In this case, there are 4 subjects. Therefore module (45) calculates the Feat value for the input graph (Feat=(5,5)) and compares it to the (11,11), which is Feat value for non-decomposable graphs:

$$Feat = (5, 5) \neq (11, 11).$$

Therefore graph is decomposable. If graph is non-decomposable, module (45) terminates RGT inference and notifies user that there is no solution, since the graph is non-decomposable. This dashed-line arrows in Fig. 13 indicate that module (45) does not terminate the RGT inference.

[0072] Module (45) takes pairs of relationships as input from (43). Then NoA (x_s , where x is either a, b, c, d) for each subject is calculated: $a_s=0$, $b_s=1$, $c_s=0$ and $d_s=1$. Using $a_s=0$, $b_s=1$, $c_s=0$ and $d_s=1$, the Feat value is calculated: Feat=(5,5). Feat value (5,5) is sent to module (49+410).

[0073] Module (41) sends NS=3 value to module (47). Module (48) contains LUT, which finds a reference to a particular LUT in module (410). The operation of selecting a record from LUT and sending reference (ref2) to the module (410) is indicated as disjunction of the modules (47) and (48) into one module (47+48). Module (47+48) sends a reference ref21 to the module (410).

[0074] Module (411) receives inputs from modules (43) outputs Feat1 (NoA) value for each subject variable to module (413).

[0075] The module (413) receives two inputs from modules (410), (411) and (412). The input from module (49+410) is used to select a certain LUT in module (412). For simplicity of presentation, only LUT corresponding to the ref21 is shown in module (412). In such case, subj1 = b, subj2 = d, subj3 = a and subj4 = c. The input from module (411) is then used to related actual subject variables and TDIs. The resulting

decision intervals for each subject are shown in module (413).

- [0076] Finally, module (414) takes inputs from Influence module (44) and module (413). Module (414) calculates values of limits of decision intervals, using the information about mutual influences from module (44). Module (414) outputs solution of RGT inference.

Industrial Applicability

- [0077] The present invention assists the Reflexive Game Theory (RGT) inference by allowing the user to input data required for conventional RGT inference.

Another application is to provide analysis of the group behavior based on the relationships between the subjects in a group and their mutual influences.

Reference Signs List

- [0078] 1, 2, 3, 4 Module

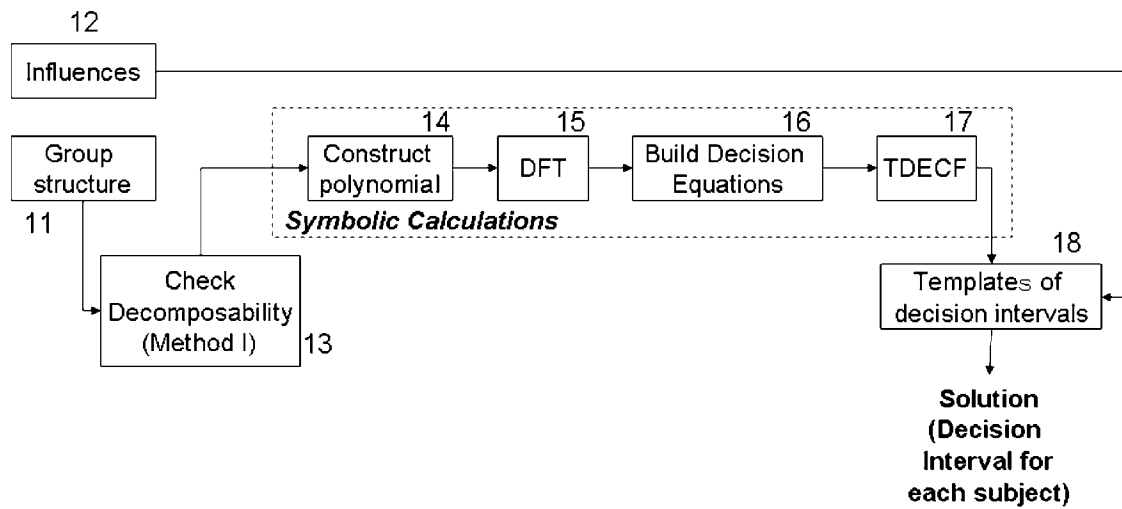
Claims

- [Claim 1] A method of performing Reflexive Game Theory (RGT) inference comprising:
pre-computing of possible group structure and real-time selecting (template matching) of the group templates by using the input information.
- [Claim 2] A method according to claim 1,
wherein the pre-computing process pre-computes the templates for template matching-based Reflexive Game Theory (RGT) inference based on using templates of the groups, and the templates are obtained by generalization of the possible group structures.
- [Claim 3] A method according to claim 2,
wherein the combinatorial techniques are applied to obtain pre-computed templates, and the combinatorial technique are applied to obtain all possible different group types.
- [Claim 4] A method of processing the actual input data for RGT inference by using pre-computed templates comprising:
identifying a certain template corresponding to the input data by unique structure features; and
identifying location of each subject by means of position feature.
- [Claim 5] The method according to claim 4,
wherein algebraic features are computed for identification of a corresponding group template.
- [Claim 6] The method according to claim 4,
wherein algebraic features are computed for identification of a location of each subject in a group.
- [Claim 7] A system of performing Reflexive Game Theory (RGT) inference comprising:
pre-computing module that pre-computes possible group structure, and
real-time selecting module that real-time selects (template matching) the group templates by using the input information.
- [Claim 8] A system according to claim 7,
wherein the pre-computing module pre-computes the templates for template matching-based Reflexive Game Theory (RGT) inference based on using templates of the groups, and the templates are obtained by generalization of the possible group structures.
- [Claim 9] A system according to claim 8,

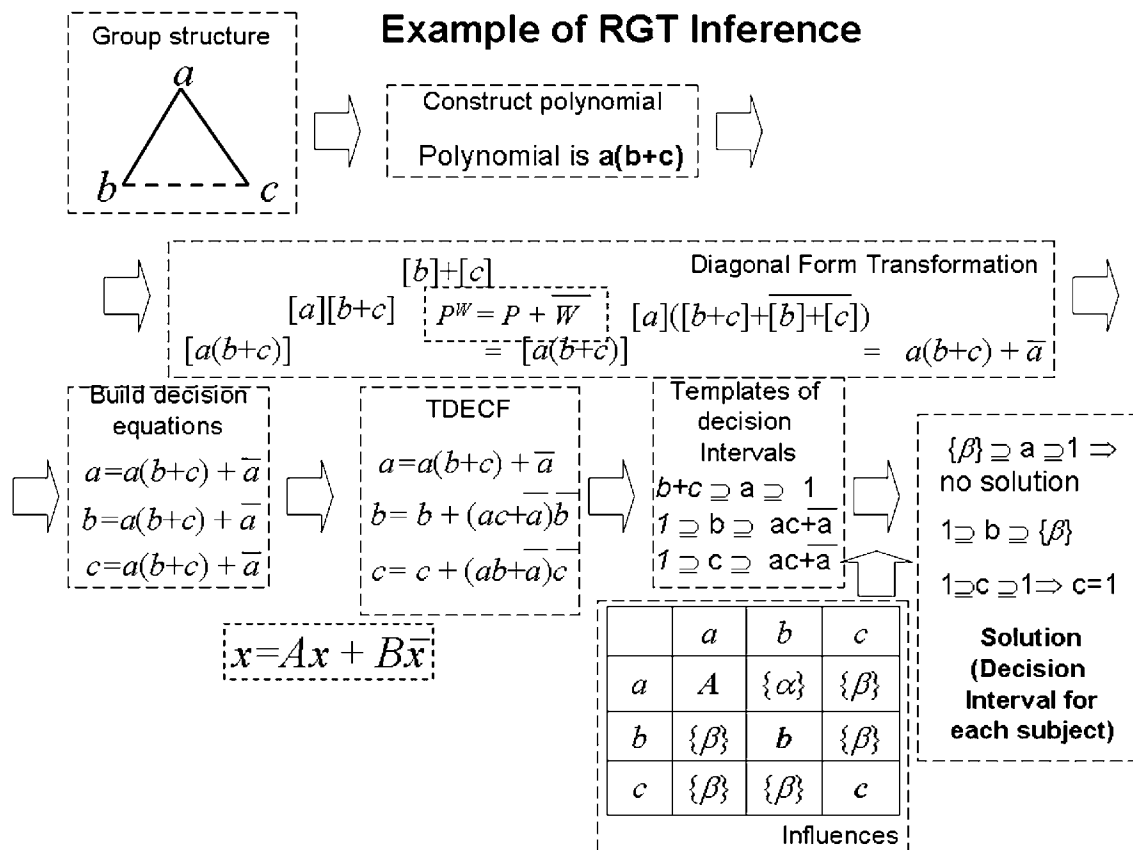
wherein the combinatorial techniques are applied to obtain pre-computed templates, and the combinatorial technique are applied to obtain all possible different group types.

- [Claim 10] A system of processing the actual input data for RGT inference by using pre-computed templates comprising:
module that identifies a certain template corresponding to the input data by unique structure features; and
module that identifies location of each subject by means of position feature.
- [Claim 11] A system according to claim 10,
wherein the module computes algebraic features for identification of a corresponding group template.
- [Claim 12] The system according to claim 10,
wherein he module computes algebraic features for identification of a location of each subject in a group.

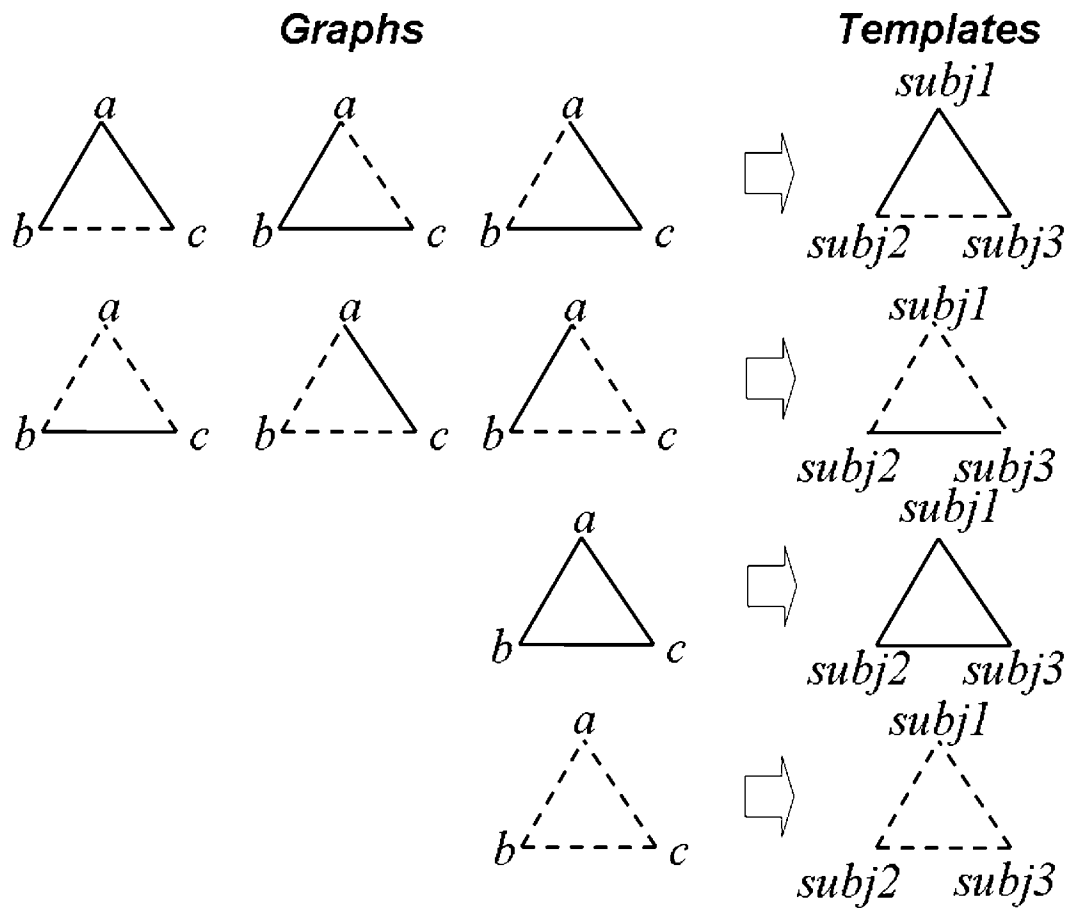
[Fig. 1]



[Fig. 2]

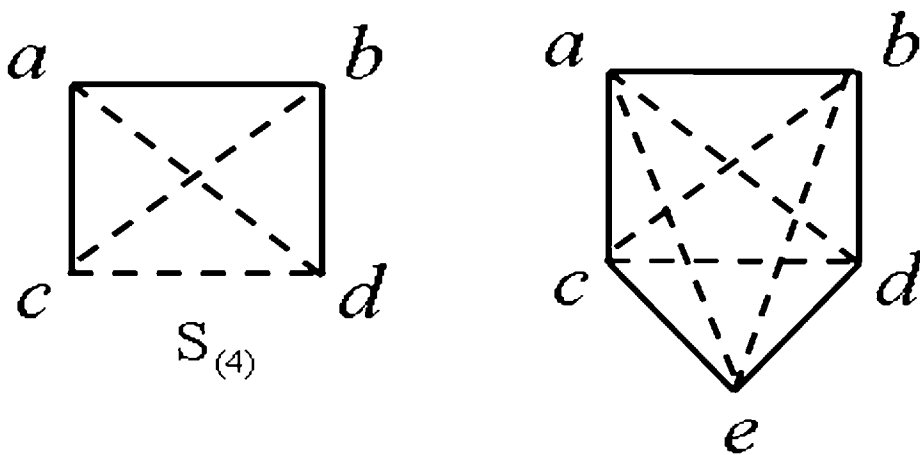


[Fig. 3]



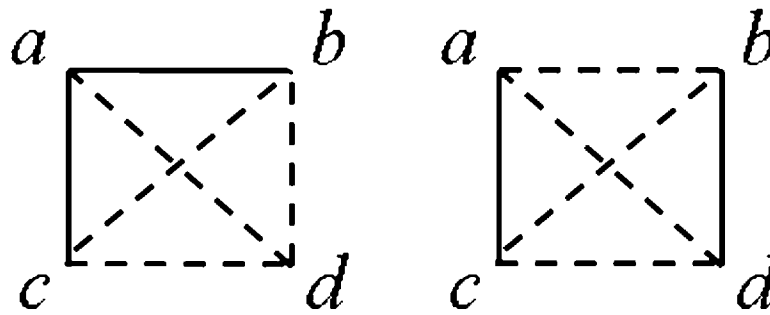
Graphs and Templates

[Fig. 4]



Examples of non-decomposable graphs

[Fig. 5]

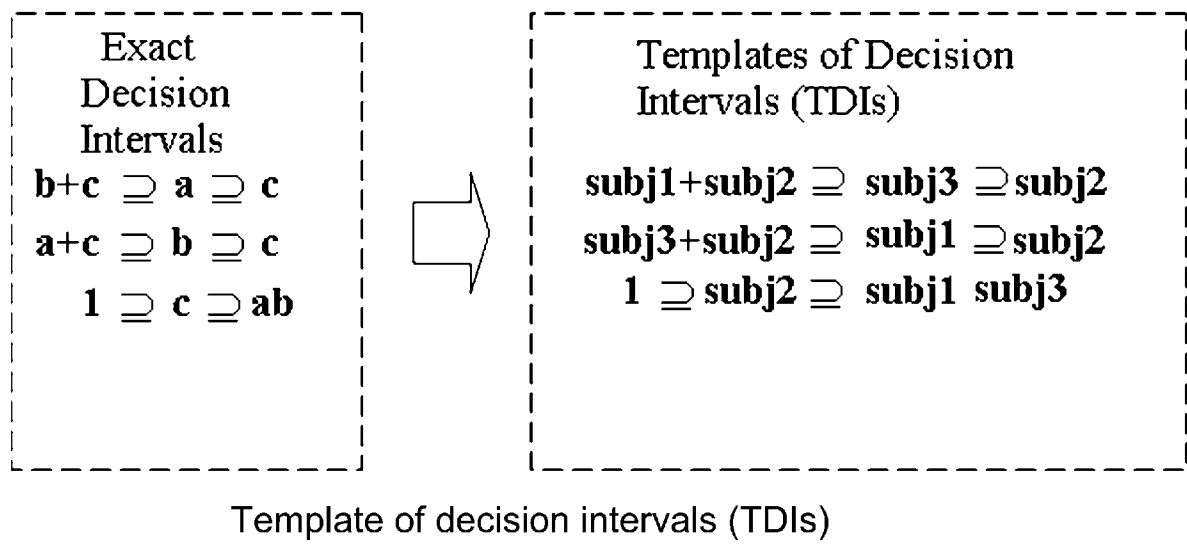
Examples of sub-classes for class $CI(k=2, n=4)$

[Fig. 6]

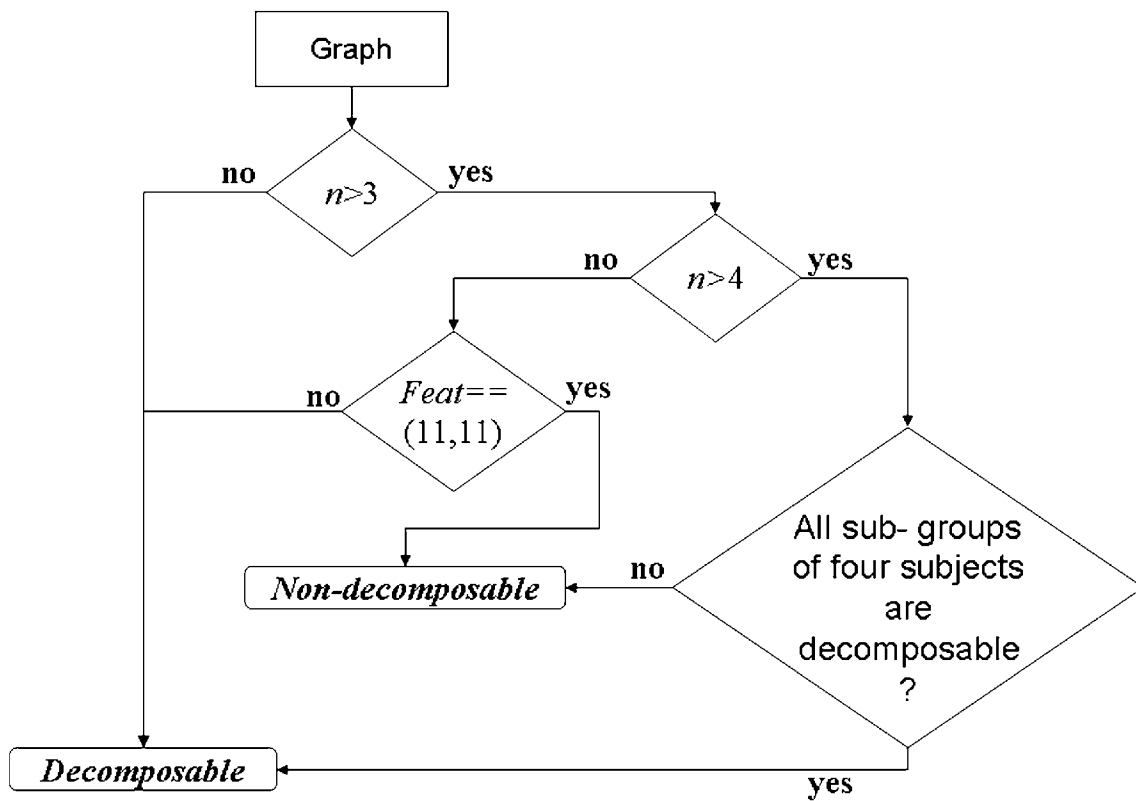
Example Graphs	Template for Graph Types	4-dim vectors	
		(0,0,0,0)	Class $CI(0,4)$ [Type1]
		(1,0,0,1)	Class $CI(1,4)$ [Type2]
		(2,1,0,1)	Sub-class 1 [Type3]
		(1,1,1,1)	Sub-class 2 [Type4]
			Class $CI(2,4)$
		(2,1,1,2)	(Non-decomposable) Sub-class 1 [Type5]
		(3,1,1,1)	Sub-class 2 [Type6]
		(2,2,0,2)	Sub-class 3 [Type7]
			Class $CI(3,4)$
		(1,2,3,2)	Sub-class 1 [Type8]
		(2,2,2,2)	Sub-class 2 [Type9]
			Class $CI(4,4)$
		(2,3,3,2)	Class $CI(5,4)$ [Type10]
		(3,3,3,3)	Class $CI(6,4)$ [Type11]

Example of Generalization result for the group of four members

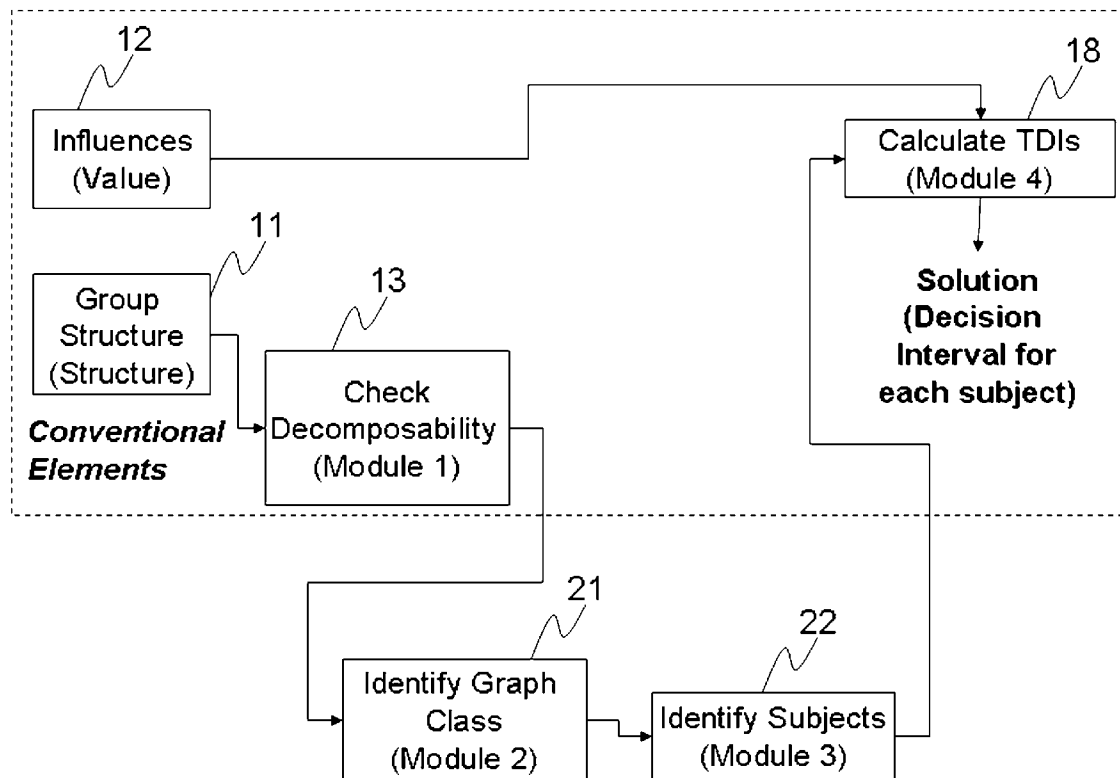
[Fig. 7]



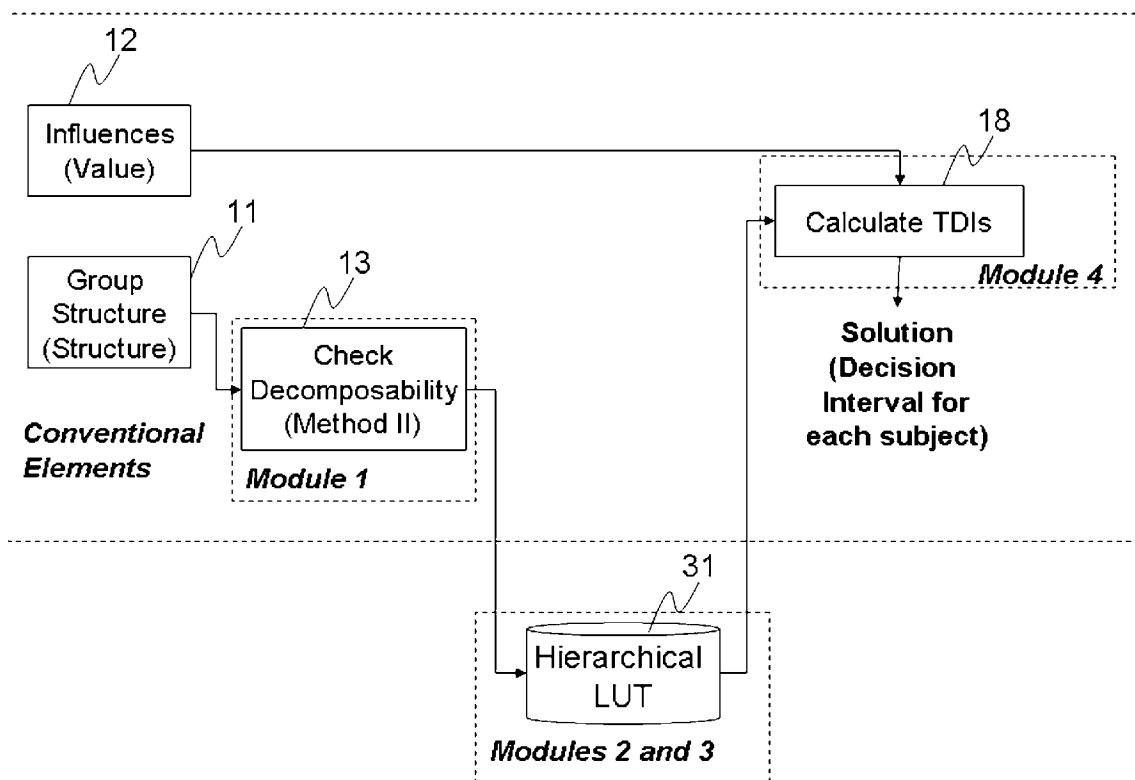
[Fig. 8]



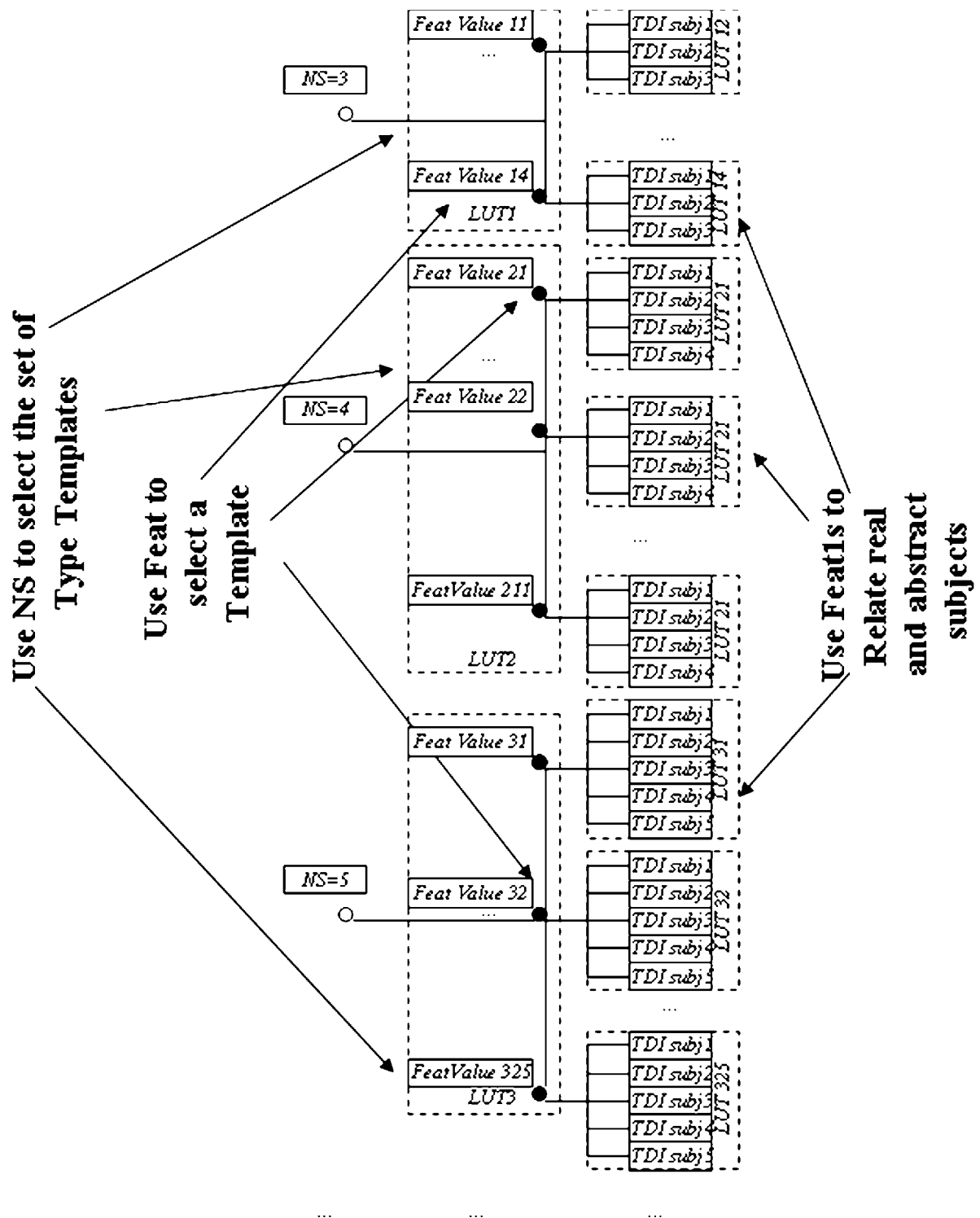
[Fig. 9]



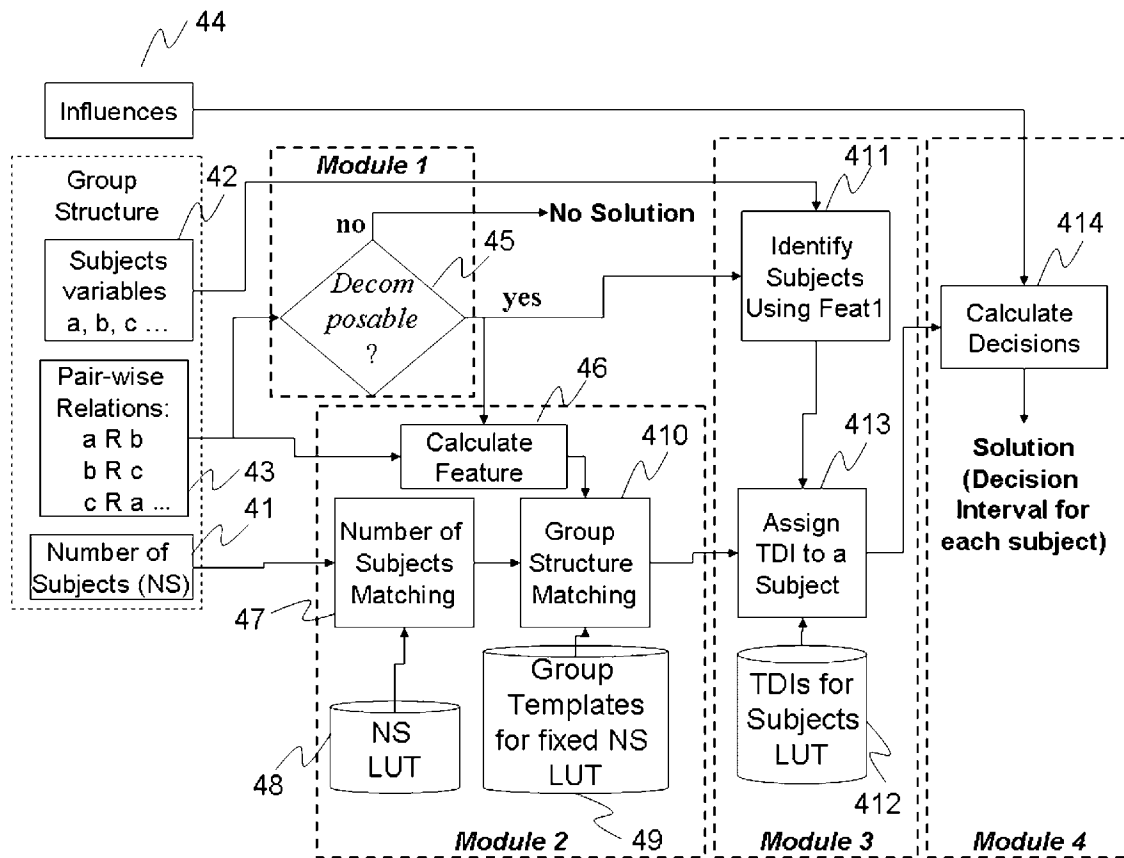
[Fig. 10]



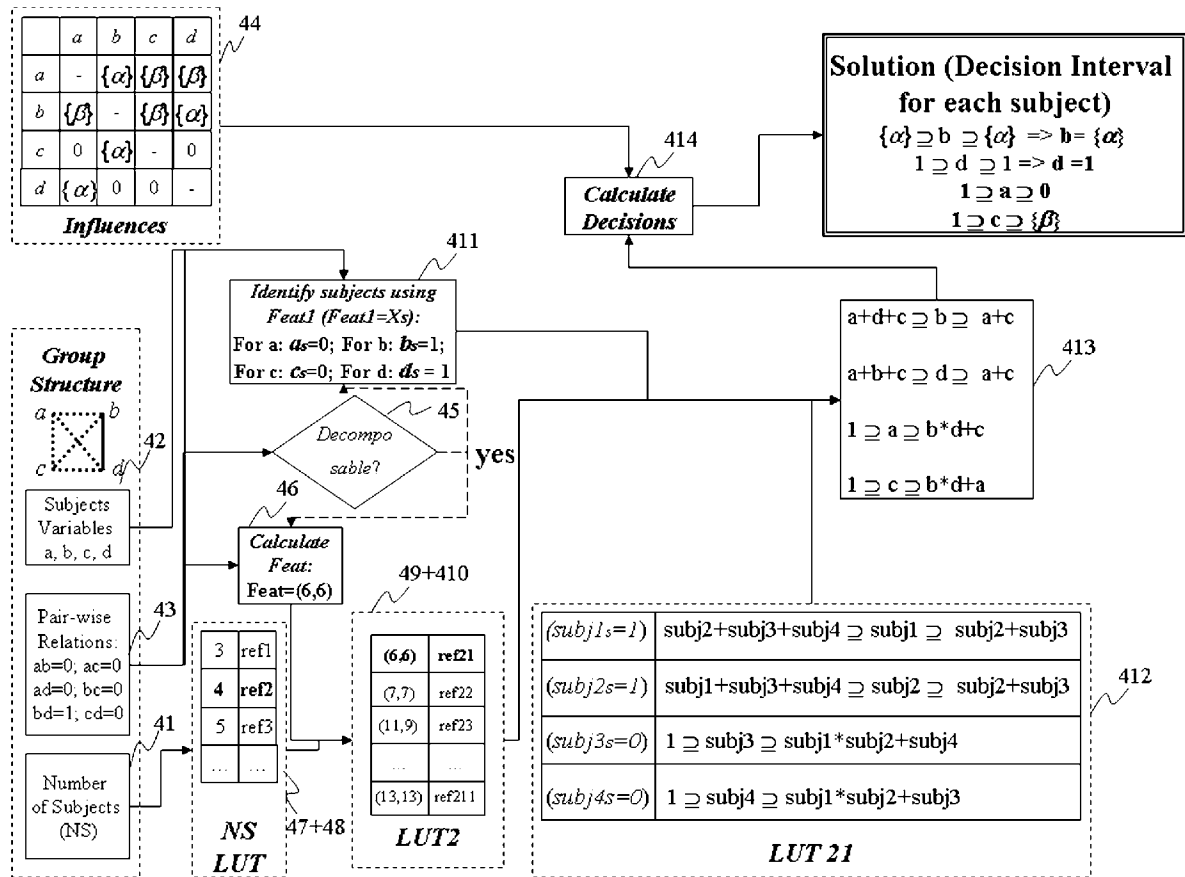
[Fig. 11]



[Fig. 12]



[Fig. 13]



INTERNATIONAL SEARCH REPORT

International application No
PCT/JP2012/005351

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06N5/00
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Sergey Tarasenko: "Modeling multistage decision processes with Reflexive Game Theory", CoRR abs/1203.2315, 11 March 2012 (2012-03-11), XP055060571, Retrieved from the Internet: URL: http://arxiv.org/pdf/1203.2315v1.pdf [retrieved on 2013-04-22] the whole document ----- -/--	1-12



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

23 April 2013

Date of mailing of the international search report

02/05/2013

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Valencia, Erika

INTERNATIONAL SEARCH REPORT

International application No

PCT/JP2012/005351

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Sergey Tarasenko: "The Inverse Task of the Reflexive Game Theory: Theoretical Matters, Practical Applications and Relationship with Other Issues", CoRR abs/1011.3397, 31 December 2010 (2010-12-31), XP55060661, Retrieved from the Internet: URL:http://arxiv.org/pdf/1011.3397v1.pdf [retrieved on 2013-04-23] the whole document -----	1-12