SEQUENTIAL KNOWLEDGE STRUCTURE IN DISTRIBUTED SYSTEM WITH AWARENESS

Susumu Yamasaki
Department of Computer Science, Okayama University, 3-1-1 Tsushima-Naka, Okayama, Japan

Keywords: Knowledge structure, Awareness, State transition.

Abstract: This position paper deals with a formal system to manage sequential knowledge structure for Web site page analyses in a distributed system, by means of the rule-based state transition. Agent technology in AI (of modern approach), logic and database for relations between action and knowledge, process algebra and related logic with respect to distributed environments, and structural analyses (which may be static, interactive or constrained) of referential knowledge for Web site pages may be relevant, however, the present paper is concerned with the sequences like Web site page ones abstracted for a model of content retrievals through communications among sites via the internet. Awareness depending of states on the communications between sites may be adopted so that sequential knowledge acquisition and management would be possibly available.

1 INTRODUCTION

We make analyses on objective knowledge: As an example of objective knowledge, the Web site page contains varieties of meanings. Exploring Web site pages, the sequence makes sense which is often related to state transitions caused by page contents. Under the constraint of state transitions, we may be interested in an automated implementation to form a sequence composed of subsequences generated by distributed calculi. For a model of forming sequence of objective knowledge, we assume a calculus as well as a managing scheme based on awareness of reasonable communications between calculi.

This paper is concerned with a model of forming sequences of objective knowledge. It is a motive to automate sequence formations, where:

(i) the sequence is composed of subsequences generated by distributed calculi, and

(ii) the constraint on state transitions for the formation comes from awareness of a calculus.

To see sequential knowledge induced by Web pages and to reach a formal system with some implementable procedures for content retrievals, we pay attention to relevance to some established frameworks:

(1) Agent technologies as compiled (Russell, 1995) may fit descriptions of knowledge sequence formation.

(2) Logic and database views are fundamental to analyze knowledge structure (Minker, 1987; Shepherdson, 1987), to understand dynamic structure with reference to knowledge (Mosses, 1992; Reiter, 2001), and to make use of distributed negatives.

(3) Process algebra is concerned with sequence structure of communications (evaluations) (Hoare, 1985; Milner, 1989) even for distributed systems (Bruns, 1996), while its logical formulation is discussed (Kucera and Esparza, 2003) such that logical frameworks may conceive modality and nomination (Areces and Blackburn, 2003; Brauner, 2004).

(4) References of Web site pages may be examined from static link view (Yamasaki, 2009) and interactive mechanism with constraints (Yamasaki, 2007).

We can observe a way of knowledge acquisition, in terms of Web site pages. Firstly, the Web site page denotes a relation between states. Secondly, as regards a visit sequence to Web site pages, we abstract a sequential structure such as:

$$\sigma_1, P_1, \sigma_2, \ldots, \sigma_n, P_n, \sigma_{n+1} \quad (n > 0),$$

where:

(i) $P_1, \ldots, P_n$ are pages, and

Yamasaki S..
SEQUENTIAL KNOWLEDGE STRUCTURE IN DISTRIBUTED SYSTEM WITH AWARENESS.
DOI: 10.5220/0003677402930298
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2011), pages 293-298
ISBN: 978-989-8425-80-5

(ii) σ_1, ..., σ_n are to denote states abstracted from knowledge base.

Thirdly, we see structure of Web site pages such that:

- A page contains a list of references to pages.

As regards the description of recursive links, a sequence of pages included in a given page is taken in this paper rather than a set of them, such that a sequential knowledge structure is studied. Such a recursive structure forms constraints of visiting pages as well as state transition sequences. We can, in what follows, construct a calculus to realize sequential structures based on the above 3 points.

For a distributed system containing calculi (which is defined in this paper), we see that *awareness* (Agotnes and Alechina, 2007) can manage the connection of calculi with communication channels, to compose subsequences generated by local calculi.

2 A MODEL FOR KNOWLEDGE ACQUISITION

To model an acquisition scheme of keywords possibly for Web usabiblity, we have considered the case that the keywords contain both positive and negative informations to denote a content. In what follows, we have some itemized aspects of behaviours of the model for keywords acquisition.

- A request containing keywords supposedly searches the Web site pages involving keywords.
- The reliable response of a Web page to the request causes an enumeration of the page and an inclusion in a list of the request data.
- A managing process with a request is interactive with multi-site (of a distributed system), where each page contains a recursive link structure in a site.
- How any page supposedly responds to the request (as a program with knowledge content) is:
  if the keywords of the page are consistent with those of the request, then they are to be merged with those of the request. If the keywords of the page are inconsistent with those of the request, the keywords kept in the request are revised to be consistent with the page ones.

(The request searches a page in the sense that their keywords are mutually consistent, and also acquire consistent keywords from the page.)

We can design a whole system, consisting of a managing program, a request, and sites with their own pages:

(a) A managing program is interactive with a site through a request of keywords. When there are more than one interaction requirement of sites, only one from a site is selected, and other requirements are kept until the interaction would be over.

(b) The request is a data structure with a function acquires consistent keywords from pages in a site and to integrates consistent keywords. The keywords contained by it may be changed through visits to site pages. In each site of the system, there are pages under the site environment. Each page of a site involves a program (to make the request data consistently revised) for keywords. If the page contains consistent keywords with those of the request, it is regarded as reliable. Otherwise, the request may be consistently revised.

We can observe the *state* denoted by keywords of the above data structure "request", such that we now have the structure of a formal system design and its management, abstracting the Web site visit sequence as well as knowledge acquisition. In this case, knowledge acquisition may be made by state transitions, changing situations of knowledge (which is realized by keywords).

We will have a formal system involving knowledge structure. It contains:

(i) a set of objects referring to pages.

(ii) a set of states.

(iii) a semantic function causing a state transition, assigned to each object.

(iv) a function to denote effects of an object sequence.

(v) a follower relation to represent an object sequence succeeding an object, which is regarded as a rule with constraints.

3 FORMAL SYSTEM FOR KNOWLEDGE STRUCTURE

When the copying the page from each site (a specific local place) to the internet (the common space) is allowed, the communication (the copying) between any two sites is possible. On the assumption that the copying of this direction is allowed, that is, the communication of the page transfer, we have a system in a distributed environments. Before the distributed system description, we have a formal system as follows: A system for knowledge structure is a quintuple $\Im = (O, \Sigma, Sem, Effect, R)$, where:

(i) O is a set of objects.

(ii) Σ is a set of states.

(iii) Sem: $O \to (\Sigma \to \Sigma)$ is a semantic function.

(iv) $Effect$: $O^* \to 2^{\Sigma\times\Sigma}$ is a function (to denote effects of a sequence of objects). Note that $O^* = \{x_1 \ldots x_n \mid n \geq 0, x_1, \ldots, x_n \in O\}$. The empty sequence in O^* is denoted by ε.

(v) $R \subseteq O \times O^*$ is a follower relation, where $(x,G) \in R$ means that x is followed by G.

Inference Rules by Means of the Follower Relation R:

We define the *inference rules* (1), (2) and (3), on the assumption that a system

$$\Im = (O, \Sigma, Sem, Effect, R)$$

is given. Note that the inference rule is defined by a scheme of "assumptions vs. conclusion" as in proof theory. That is, we have the notation: Assumptions (of the form "$Pr_1 \ldots Pr_n$" to be connected by "and") versus Conclusion, like the inference rule.

(1) $$\frac{}{move_R(\varepsilon;\sigma;\sigma)}$$

(2) $$\frac{(x,G) \in R \quad Sem[\![x]\!]\sigma_1 = \sigma_2' \quad move_R(G;\sigma_2';\sigma_2)}{move_R(x;\sigma_1;\sigma_2)}$$

(3) $$\frac{move_R(G_1;\sigma_1;\sigma_2) \quad move_R(G_2;\sigma_2;\sigma_3)}{move_R(G_1G_2;\sigma_1;\sigma_3)}$$

The Meaning of the Relation $move_R$

Intuitively, the relation $move_R \subseteq O^* \times \Sigma \times \Sigma$ is defined, such that by $move_R(\gamma;\sigma_1;\sigma_2)$, we mean that:

- Given the sequence γ initiated, the state transition from σ_1 to σ_2 is caused by rewriting (owing to the follower relation) and reducing γ to the empty sequence.

We have accounts for the above inference rules.

(i) The empty sequence ε causes the empty state transition.

(ii) If there is $(x,G) \in R$ such that $Sem[\![x]\!]\sigma_1 = \sigma_2'$ and the sequence G causes a state transition $\sigma_2' \to \sigma_2$ (from σ_2' to σ_2), then the object x causes a transition $\sigma_1 \to \sigma_2$ (from σ_1 to σ_2).

(iii) For the sequences G_1 and G_2, respectively causing the transitions from σ_1 to σ_2 and from σ_2 to σ_3, the sequence G_1G_2 causes the transition from σ_1 to σ_3.

We denote the derivation of the predicate $move_R(G;\sigma_1;\sigma_2)$ with applications of the inference rules (1)-(3) finitely many times, by the notation "$move_R(G;\sigma_1;\sigma_2)$" itself.

The Meaning of the Expression $Effect$

In accordance with the inferences, the effects of an object sequence are defined as follows. Note that $Effect[\![\beta]\!](\sigma_1,\sigma_2)$ stands for $(\sigma_1,\sigma_2) \in Effect[\![\beta]\!]$. The relation $Effect[\![\beta]\!]$ means that if (σ_1,σ_2) is included in, it denotes a possible transition from σ_1 to σ_2 by means of a sequence β. That is, β is an effective sequence for the transition.

(1) $$\frac{}{Effect[\![\varepsilon]\!](\sigma,\sigma)}$$

(2) $$\frac{(x,G) \in R \quad Sem[\![x]\!]\sigma_1 = \sigma_2' \quad Effect[\![G]\!](\sigma_2';\sigma_2)}{Effect[\![xG']\!](\sigma_1;\sigma_2)}$$

(3) $$\frac{Effect[\![G_1]\!](\sigma_1;\sigma_2) \quad Effect[\![G_2]\!](\sigma_2;\sigma_3)}{Effect[\![G_1G_2]\!](\sigma_1;\sigma_3)}$$

The following propositions are regarded as the special cases of corresponding Corollary 1 and Theorem 3 in distributed systems.

We have the following proposition between the transition expressed by $move_R$ and $Effect[\![-]\!]$.

Proposition 1.

$\exists G.[move_R(G;\sigma_1;\sigma_2)]$ iff $\exists F.[Effect[\![F]\!](\sigma_1;\sigma_2)]$.

We have a procedure to extract a real sequence to cause the state transition:

Object-sequence Formation for $\Im$:

$formation(G;\sigma_1;\sigma_2) \Leftarrow$
if $G = \varepsilon$
then **if** $\sigma_1 = \sigma_2$ **then** ε (empty sequence)
else
if $G = x$ such that $(x,G') \in R$ and $Sem[\![x]\!]\sigma_1 = \sigma_2'$
then x followed by $formation(G';\sigma_2';\sigma_2)$
else
if $G = G_1G_2$ such that
$formation(G_1;\sigma_1;\sigma_1')$ and
$formation(G_2;\sigma_1';\sigma_2)$ are defined for some σ_1'
then
$formation(G_1;\sigma_1;\sigma_1')$
followed by $formation(G_2;\sigma_1';\sigma_2)$

We can have the proposition between the effect of procedure $formation$ and $Effect[\![-]\!]$:

Proposition 2. $\exists G.[formation(G;\sigma_1;\sigma_2)$ provides $F]$ iff $Effect[\![F]\!](\sigma_1;\sigma_2)$.

4 DISTRIBUTED SYSTEM FOR KNOWLEDGE STRUCTURE

We now deal with the distributed system consisting of calculi (as in the previous section) where the communications between calculi are free so that the object in the set O can be transferred from one calculus to another.

A Distributed System

In what follows, we have a formal system to involve an effective sequence for each calculus, to be extracted. Now a distributed system (for knowledge structure) is an n-tuple

$$DS = <\Im_1,\ldots,\Im_n;\mathcal{A}> (n \geq 1),$$

where each $\Im_i$ is a system $(O,\Sigma,Sem_i,Effect_i,R_i)$ for knowledge structure, and *awareness* $\mathcal{A}$ is defined as follows. We here assume a mapping

$$\mathcal{A} : \Sigma \rightarrow \{\text{receive}_i,\text{send}_j \mid 1 \leq i, j \leq n\},$$

where if we have send_j, $\text{receive}_i \in \mathcal{A}(\sigma)$, then it is described by:

$$j \rightarrow_\sigma i,$$

which means that there may be a communication from the calculus $\Im_j$ to $\Im_i$ through the state in Σ. Note that we may see details of awareness (Agotnes and Alechina, 2007). It can be supposed that $i \rightarrow_\sigma i$ for any $\sigma \in \Sigma$ and for any $\Im_i$. That is, $\rightarrow_\sigma$ is reflexive, which is implicitly included in the following inference rules.

The inference rule of the provious section for $move_R$ may be generalized to the one, $move_{R_i}$ $(1 \leq i \leq n)$. The relation $move_{R_i}$ of the system $\Im_i$ involves the usage of the function Sem_j of the system $\Im_j$.

Inference Rules for $\Im_i$ by Means of R_i:

(1) $$\frac{}{move_{R_i}(\varepsilon;\sigma;\sigma)}$$

(2) $$\frac{(x,G) \in R_i \; Sem_j[\![x]\!]\sigma_1 = \sigma_2' \; j \rightarrow_{\sigma_2'} i \; move_{R_i}(G;\sigma_2';\sigma_2)}{move_{R_i}(x;\sigma_1;\sigma_2)}$$

(3) $$\frac{move_{R_i}(G_1;\sigma_1;\sigma_2) \;\; move_{R_i}(G_2;\sigma_2;\sigma_3)}{move_{R_i}(G_1G_2;\sigma_1;\sigma_3)}$$

In accordance with the inference, we might have the rules for effects of object sequences.

The relation $Effect[\![-]\!]$ is extended to the one, $Effect_i[\![-]\!]$. Different from the relation $Effect[\![-]\!]$, the relation $Effect_i[\![-]\!]$ is concerned with the sequence caused only by the follower relation of the system $\Im_i$, but not any sequence caused by other systems $\Im_j$ $(\imath \neq j)$.

Constructive Definition of $Effect_i$

(1) $$\frac{}{Effect_i[\![\varepsilon]\!](\sigma,\sigma)}$$

(2) $$\frac{(x,G) \in R_i \;\; Sem_j[\![x]\!]\sigma_1 = \sigma_2' \;\; j \rightarrow_{\sigma_2'} i \quad Effect_i[\![F]\!](\sigma_2';\sigma_2)}{Effect_i[\![xF]\!](\sigma_1;\sigma_2)}$$
$(i = j)$

$$\frac{(x,G) \in R_i \;\; Sem_j[\![x]\!]\sigma_1 = \sigma_2' \;\; j \rightarrow_{\sigma_2'} i \quad Effect_i[\![F]\!](\sigma_2';\sigma_2)}{Effect_i[\![F]\!](\sigma_1;\sigma_2)}$$
$(i \neq j)$

(3) $$\frac{Effect_i[\![F_1]\!](\sigma_1;\sigma_2) \;\; Effect_i[\![F_2]\!](\sigma_2;\sigma_3)}{Effect_i[\![F_1F_2]\!](\sigma_1;\sigma_3)}$$

Effective Sequence from the Relation $move_{R_i}$

The following theorem suggests that we can have an effective sequence F with $Effect_i[\![F]\!]$ on the basis of the relation $move_{R_i}$. The proof is presented in Appendix.

Theorem 1. On the assumption of a distributed system (for knowledge structure) is an n-tuple

$$DS = <\Im_1,\ldots,\Im_n;\mathcal{A}> \;\; (n \geq 1),$$

where each $\Im_i$ is a system $(O,\Sigma,Sem_i,Effect_i,R_i)$ for knowledge structure, suppose that $move_{R_i}(G;\sigma_1;\sigma_2)$. Then there is a sequence F such that $Effect_i[\![F]\!](\sigma_1,\sigma_2)$.

The Relation $Move_{R_i}$ caused by $Effect_i[\![-]\!]$

The following theorem suggests that we can have a relation $move_{R_i}$ on the basis of the relation $Effect_i[\![-]\!]$. The proof is in Appendix.

Theorem 2. If $Effect_i[\![F]\!](\sigma_1,\sigma_2)$ then there is a sequence G such that $move_{R_i}(G;\sigma_1;\sigma_2)$.

Equivalence between $move_{R_i}$ and $Effect_i[\![-]\!]$

By Theorems 1 and 2, we have:

Corollary 1. There is a sequence G such that $move_{R_i}(G;\sigma_1;\sigma_2)$ iff there is a sequnce F such that $Effect_i[\![F]\!](\sigma_1,\sigma_2)$.

We next have a distributed procedure:

The procedure can demonstrate a sequence to cause a transition between two given states.

Object-sequence Formation for $\Im_i$ in *DS*:

$d\text{-}formation_i(G;\sigma_1;\sigma_2) \Leftarrow$
if $G = \varepsilon$
  then
    if $\sigma_1 = \sigma_2$ **then** ε (empty sequence)
  else
    if $G = x$ such that $(x, G') \in R_i$, $j \rightarrow_{\sigma_2'} i$,
        and $Sem_j[\![x]\!]\sigma_1 = \sigma_2'$
    then
    if $(i = j)$
      then x followed by $d\text{-}formation_i(G';\sigma_2';\sigma_2)$
      else $d\text{-}formation_i(G';\sigma_2',\sigma_2)$
    else
      if $G = G_1G_2$ such that
        $d\text{-}formation_i(G_1;\sigma_1;\sigma_1')$ and
        $d\text{-}formation_i(G_2;\sigma_1';\sigma_2)$ are defined
      for some σ_1'
        then
        $d\text{-}formation_i(G_1;\sigma_1;\sigma_1')$
        followed by $d\text{-}formation_i(G_2;\sigma_1';\sigma_2)$

The following theorem, whose proof is given in Appendix, states the effect(s) by the procedure $d\text{-}formation_i$ as well as in terms of $Effect_i[\![-]\!]$.

Theorem 3. *$d\text{-}formation_i(G;\sigma_1;\sigma_2)$ provides F for some G iff $Effect_i[\![F]\!](\sigma_1;\sigma_2)$.*

We can have concluding remarks on the distributed system where:

(i) Which calculi are a pair of a sender and a receiver is controlled by awarenes of states. This notion is not only a software method but an AI tool, if this formal system is applicable to a model of a part of brain works for sequence knowledge.

(ii) The abstract notions whcih we have presented, $move_{R_i}(G;\sigma_1;\sigma_2)$, $Effect_i[\![F]\!](\sigma;\sigma_2)$ and $d\text{-}formation_i(G';\sigma_1;\sigma_2)$, are concerned with the state transition from σ_1 to σ_2, the extraction of an effective sequence only from the calculus $\Im_i$, and a sequence construction in the calculus $\Im_i$, respectively .

The system is an abstract scheme to contribute to a formation of sequences composed of distributed subsequences, free from more specific e-Learning mechanism (Sasakura and Yamasaki, 2008) ane eventformation (Yamasaki and Sasakura, 2008).

REFERENCES

Agotnes, T., Alechina, N. (2007). Full and relative awareness: A decidable logic for reasoning about knowledge of unawareness. In *Theoretical Aspects of Rationality and Knowledge, Brussels*, 6–14.

Areces,C. and Blackburn,P. (2003). Repairing the interpolation in quantified logic. *Annals of Pure and Applied Logic 123*, 287–299.

Brauner,T. (2004). Natural deduction for hybrid logics. *J. of Logic and Computation 14*, 329–353.

Bruns,G. (1996). *Distributed Systems Analysis with CCS*. Prentice-Hall.

Genesereth,M.R. and Nilsson,N.J. (1988). *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann.

Hoare,C.A.R. (1985). *Communicating Sequential Processes*. Prentice Hall.

Kucera,A. and Esparza,J. (2003). A logical viewpoint on process-algebra, *J. of Logic and Computation 13*. 863–880.

Milner,R. (1989). *Communication and Concurrency*. Prentice Hall.

Minker,J. (ed.) (1987). *Foundations of Deductive Databases and Logic Programming*. Morgan Kaufmann Publishers Inc.

Mosses,P.M. (1992). *Action Semantics*. Cambridge University Press.

Reiter,R. (2001). *Knowledge in Action*. The MIT Press.

Russell,S. and Norvig,P. (1995). *Artificial Intelligence–A Modern Approach–*. Prentice Hall.

Sasakura,M. and Yamasaki,S. (2008) A graph manipulation visual interface for construction of e-Learning systems. *Proc. of 12th International Conference on Information Visualization*, 644–649.

Shepherdson,J.C. (1987). Negation in logic programming. In *Minker,J. (ed.), Foundations of Deductive Databases and Logic Programming*, 19–88.

Yamasaki,S. (2007) An interactive constraint system and its relation to logic in AI. *Proc. of 19th Belgium/Netherlands Artificial Intelligence Conference*, 287–294.

Yamasaki,S. and Sasakura,M. (2008) A calculus effectively performing event formations with visualization. *Lecture Notes in Computer Science 4759*, 287–294.

Yamasaki,S. (2009) A fixed point representation of references. *University of Oslo Department of Informartics, Research Report 386*, 120–129.

APPENDIX

Proof of Theorem 1:
It is proved by structural induction on the construction of the predicate $move_{R_i}(G;\sigma_1;\sigma_2)$.

(a) In case that $G = \varepsilon$, it follows for the assumed predicate $move_{R_i}(G;\sigma_1;\sigma_2)$ that $\sigma_1 = \sigma_2$. Then $Effect_i[\![\varepsilon]\!](\sigma_1,\sigma_2)$ where $\sigma_1 = \sigma_2$.

(b) In case that $G = x \in O$, the predicate $move_{R_i}(x;\sigma_1;\sigma_2)$ is derived by means of the inference rule (2). Assume for the predicate $move_{R_i}(x;\sigma_1,\sigma_2)$ that $(x,G') \in R$, $Sem_j[\![x]\!]\sigma_1 = \sigma'_2$, $j \rightarrow_{\sigma'_2} i$ and $move_{R_i}(G';\sigma'_2,\sigma_2)$ for G'. For the predicate $move_{R_i}(G';\sigma'_2,\sigma_2)$, by induction hypothesis, there is a sequence F such that $Effect_i[\![F]\!](\sigma'_2,\sigma_2)$. By means of the rules (2) for effects: If $i = j$, then $Effect_i[\![xF]\!](\sigma_1,\sigma_2)$. If $i \neq j$, then $Effect_i[\![F]\!](\sigma_1,\sigma_2)$.

(c) In case that $G = G_1G_2$, we assume the predicate $move_{R_i}(G_1G_2;\sigma_1;\sigma_2)$. For the predicate

$$move_{R_i}(G_1G_2;\sigma_1;\sigma_2),$$

assume that $move_{R_i}(G_1;\sigma_1;\sigma'_2)$ and $move_{R_i}(G_2;\sigma'_2;\sigma_2)$ for some σ'_2. By induction hypothesis, we can see that for some F_1 and F_2,

$$Effect_i[\![F_1]\!](\sigma_1,\sigma'_2), \text{ and } Effect_i[\![F_2]\!](\sigma'_2,\sigma_2).$$

It follows from the rule (3) for effects: with F_1F_2,

$$Effect_i[\![F_1F_2]\!](\sigma_1,\sigma_2).$$

This completes the induction step.

Q.E.D.

Proof of Theorem 2:

It is proved by structural induction on a sequence G in the premise of the theorem. For the premise:

(a) While $Effect_i[\![\varepsilon]\!](\sigma_1,\sigma_1)$ holds, we independently have the predicate $move_{R_i}(\varepsilon;\sigma_1;\sigma_2)$ for $\sigma_2 = \sigma_1$, which is sufficient for the proof.

(b) Assume the case that $Effect_i[\![F]\!](\sigma_1,\sigma_2)$ by $(x,G') \in R_i$, $j \rightarrow_{\sigma'_2} i$, $Sem_j[\![x]\!]\sigma_1 = \sigma'_2$ and $Effect_i[\![F']\!](\sigma'_2,\sigma_2)$ $(F = xF')$. By induction hypothesis, $move_{R_i}(G';\sigma'_2;\sigma_2)$ for some G'. Then we have the predicate $move_{R_i}(G;\sigma_1;\sigma_2)$ by the $move_{R_i}$ definition.

(c) In case that $Effect_i[\![F_1F_2]\!](\sigma_1,\sigma_2)$. It is supported that, for some F_1 and F_2,

  (i) $Effect_i[\![F_1]\!](\sigma_1,\sigma'_1)$, and

  (ii) $Effect_i[\![F_2]\!](\sigma'_1,\sigma_2)$.

By induction hypothesis, we have both

$$move_{R_i}(G_1;\sigma_1;\sigma'_1)$$

for some G_1 and

$$move_{R_i}(G_2;\sigma'_1;\sigma_2)$$

for some G_2. It follows that $move_{R_i}(G_1G_2;\sigma_1;\sigma_2)$. This completes the induction.

Q.E.D.

Proof of Theorem 3:

It is proved by structural induction on the procedure $d\text{-}formation$ with reference to existing G.

(a) (Basis)
We see that $d\text{-}formation_i(\varepsilon;\sigma_1;\sigma_1)$ provides ε iff and $Effect_i[\![\varepsilon]\!](\sigma_1;\sigma_1)$.

(b) (Induction 1) Assume that $G = x \in O$. By the definitions of $d\text{-}formation_i$ and $Effect_i$, we see that:

  (i) $d\text{-}formation_i(x;\sigma_1;\sigma_2)$ provides F iff there is some $(x,G') \in R_i$ such that $Sem_j[\![x]\!]\sigma_1 = \sigma'_2$, $j \rightarrow_{\sigma'_2} i$ and

$$F = x.\ d\text{-}formation_i(G';\sigma'_2;\sigma_2)\ (i = j)$$
$$\text{or } F = d\text{-}formation_i(G';\sigma'_2;\sigma_2)\ (i \neq j).$$

  (ii) $Effect_i[\![F]\!](\sigma_1;\sigma_2)$ iff there is some $(x,G') \in R_i$ such that $Sem_j[\![x]\!]\sigma_1 = \sigma'_2$, $j \rightarrow_{\sigma'_2} i$ and

$$Effect_i[\![F']\!](\sigma'_2,\sigma_2) \text{ for } F = xF'\ (i = j),$$
$$\text{or } Effect_i[\![F']\!](\sigma'_2,\sigma_2) \text{ for } F = F'\ (i \neq j).$$

By induction hypothesis, for some G', $d\text{-}formation_i(G';\sigma'_2;\sigma_2)$ provides F' iff

$$Effect_i[\![F']\!](\sigma'_2;\sigma_2).$$

It follows that $d\text{-}formation_i(x;\sigma_1;\sigma_2)$ provides F iff

$$Effect_i[\![F]\!](\sigma_1;\sigma_2).$$

This completes the induction 1.

(c) (Induction 2) Assume that $G = G_1G_2 \neq \varepsilon$. By the definitions of $d\text{-}formation_i$ and $Effect_i$, we see that:

  (i) $d\text{-}formation_i(G_1G_2;\sigma_1;\sigma_2)$ provides F iff

$$F = d\text{-}formation_i(G_1;\sigma_1;\sigma'_1)$$
$$\text{followed by } d\text{-}formation_i(G_2;\sigma'_1;\sigma_2)$$

for some σ'_1.

  (ii) $Effect_i[\![F]\!](\sigma_1;\sigma_2)$ iff $F = F_1F_2$ for some F_1 and F_2 with some σ'_1, where

$$Effect_i[\![F_1]\!](\sigma_1,\sigma'_1), \text{ and } Effect_i[\![F_2]\!](\sigma'_1,\sigma_2).$$

By induction hypothesis, we see that:

$$d\text{-}formation_i(G_1;\sigma_1;\sigma'_1) \text{ provides } F_1 \text{ iff } Effect_i[\![F_1]\!](\sigma_1,\sigma'_1), \text{ and}$$
$$d\text{-}formation_i(G_2;\sigma'_1;\sigma_2) \text{ provides } F_2 \text{ iff } Effect_i[\![F_2]\!](\sigma'_1,\sigma_2).$$

It follows that $d\text{-}formation_i(G_1G_2;\sigma_1;\sigma_2)$ provides F_1F_2 iff $Effect_i[\![F_1F_2]\!](\sigma_1,\sigma_2)$. This completes the induction 2.

Q.E.D.