An object-oriented knowledge-based approach for formulating applied general equilibrium models

Zhi Wang*, a, James Slagleb

a World Development Report, The World Bank, Washington, DC 20433, USA
b Development of Computer and Information Sciences, University of Minnesota, Minneapolis, MN 55455, USA

(Received December 1993; final version received December 1994)

Abstract

This paper proposes and illustrates an object-oriented, knowledge-based approach to formulate Applied General Equilibrium (AGE) models. In this framework, an AGE model is viewed as a collection of structured objects from the real world such as economic agents and prices. A frame representation of major objects in AGE models for world trade is presented. This representation can be used as the structure of a model base in a loosely coupled intelligent modeling system. In such a system, constructing an AGE model appears to be an intuitive rather than an abstract process. This intuitive process needs more understanding of substance of economics and the logic underlying the problem at hand rather than mathematical notation.

Key words: Object-oriented concepts; Knowledge-based system; Class frames; Applied general equilibrium models

JEL classification: C68; D58; C88

1. Introduction

One of the major advances in applied economics since the 1970s is the conversion of the well-known Walrasian general equilibrium structure (formalized in the 1950s by Kenneth Arrow, Gerard Debreu, and others) from an

*Corresponding author.

abstract representation of an economy into realistic models of actual economies that can be used to conduct policy evaluations by specifying production and demand functions and incorporating data reflective of the real world. This thread begins with the earlier work by Johansen (1960) and Harberger's (1962) two-sector numerical general equilibrium policy models. It is stimulated by Scarf's (1967) fixed-point algorithm for the numerical determination of the equilibrium of a Walrasian system, and winds through Shoven and Whalley's models for tax reform in developed countries (1973, 1992) and global trade (1974, 1992) during the last two decades, to the developing country applications starting from Adelman and Robinson (1978). Hundreds of such models have been built and applied to a number of policy issues ranging from public finance and taxation, economic integration, GATT negotiations, and issues of North-South trade, to the evaluation of development strategies, and energy and environment policies over almost all the major countries in the world.

Paralleling this evolution of applied general equilibrium models (AGE) has been the development of computer-based support for the representation and solution of these models. The initial efforts for representing AGE models were problem-specific and required the construction of FORTRAN routines to represent the model in a computer-readable form and to conduct calibration and equilibrium computation. The equilibrium computations were usually based on extensions of Scarf's fixed-point algorithm or variants of the Newton method for solving systems of nonlinear equations. In those days, algorithmic and computer power were the main concern of modelers, since both the technical difficulties and computing costs were associated with the process of finding a numerical equilibrium to the model, whereas they are no longer problems today. Statistics show that the numerical solution does not account for more than 15% of the overall modeling effort. Model formulation, implementation, data preparation, and simulation results interpretation have become major burdens of the modelling process. Moreover, the AGE models built by professionals in academic institutions are still some distance from the kind of practical, implementable policy evaluation tools that can be easily understood and used by policy makers in the government sector. Most policy makers even do not trust the results from such models since they are not sure what economic assumptions were made in developing the models. If a policy maker is interested in using the model, a technical staff is needed to bridge the gap. The extensive time and money requirements may prohibit many policy makers from even attempting to satisfy their own curiosity with regard to the models. The value of a model may decrease to zero within a few years, since one of the key staff members changes jobs. This means that many models previously developed are not reusable.

The difficulties come from the fact that three representations of the same model are needed before any policy simulation can be carried out: (1) a semantic representation which describes the policy problem and the model to address it in
intuitive terms and that can be understood by decision makers; (2) a formal representation in which the intuitive model is transformed into a formal AGE (mathematical) model by skilled analytical professionals;\(^1\) (3) and a computer and algorithm representation which is usually implementation-specific. Any formal model needs a computer-readable form that is acceptable to an algorithm, and algorithms need a data structure and a problem representation, while problem representations that are meaningful to humans are not acceptable to machines. The three-way translation process is the major source of difficulties and errors.

Previous efforts to overcome those difficulties have been mainly along two lines. One is the development of Modeling Language Systems such as GAMS (Brooke et al., 1988), MPS/GE (Rutherford, 1989), and GEMPACK (Codsi and Pearson, 1988). By facilitating the algebraic description of the AGE models in computer-readable form and providing ready access to relational data base capability, these modeling languages implement large AGE models much more quickly and cheaply than before – assuming the models have been formulated. Their principal goal attempt is to overcome the problems associated with the translation between the model's formal and computer representations. They clearly show a tendency to move away from representations required by algorithms to representations closer to the modeler’s conception of the problem (Krishnan et al., 1988). Input to these modeling systems can also be understood by others who are familiar with mathematical notation. However, the details of a model in such a system must be derived by the modeler with paper and pencil and entered into computers by hand. It is easy to make mistakes while doing manual derivations or in changing model specifications. A sign may be wrong, or an equation can be entered incorrectly. The difficulties and problems from the translation between a model’s formal and computer representation were reduced, but still remain (Drud, 1989).

This led to the second line of research in constructing a modeling system with domain-specific knowledge such as HERCULES (Drud, 1989). HERCULES used GAMS as its modeling language and had a knowledge base about static AGE specifications to facilitate the formulation of Social Accounting Matrix-based AGE models. Users of HERCULES need only specify the economic agents included in the analysis and the functional forms of the production and demand functions. They need not be able to develop and maintain the algebraic format of AGE models. The rules in its knowledge base distill much of the knowledge about AGE modeling accumulated at the World Bank in recent years.

\(^1\)These models are often difficult to understand by policy markers who are not familiar with mathematical notation.
Although this knowledge-based approach is still in its early stages of development, it is consistent with the general trend of moving away from modeling-professional-oriented to end-user-oriented interfaces in most recently developed computer-aided model construction systems. It clearly shows that in order to overcome the difficulties and eliminate the errors caused from the three-way translation among different model representations, and to make model and modeling knowledge reusable, a knowledge-based component is essential for any new generation of economic modeling systems. The development of a modeling knowledge-base is expensive. However, once formed, it can be shared by many naive users as well as by modeling experts, permitting them to concentrate their time on economic issues and the logic of the problem rather than on technical details.

These modeling system developments facilitated the application of the AGE model in policy analysis. However, a unified conceptual framework is needed to guide those efforts. The dominant paradigm in computer science has increasingly become object-oriented in recent years, but no one has tried applying this new approach to develop an AGE modeling system. The reason is that most economists working in model development are not familiar with, and even not aware of the ongoing fundamental change in computer technology. Similarly, computer scientists are usually not familiar with domain-specific knowledge such as economics. It would be apparent that AGE modeling is an ideal case for applying the object-oriented approach. The introduction of object-oriented concepts and knowledge representation into the AGE modeling process will greatly facilitate all the modeling tasks such as interfacing the model with users, constructing and manipulating models or components of models, and interpreting model outputs.

In this paper, we propose an object-oriented knowledge-based approach to formulate AGE trade models and discuss what knowledge an AGE modeling knowledge-base should possess, how should it be represented and implemented. The paper is organized into 5 sections. Section 2 outlines core object-oriented concepts and presents an intuitive object-oriented view of AGE models. Based on such a view, Section 3 presents a formal object-oriented representation of AGE world trade models based on frames, and Section 4 outlines the architecture of an object-oriented knowledge-based AGE modeling system and briefly discusses the implementation strategies. Finally, the concluding remarks are presented in Section 5.

2. Object-oriented approach and its view of AGE models

Although the object-oriented approach has its roots in programming language, it is more than a methodology of programming. Its advantage and usefulness for software engineering, data modeling, information system analysis
and design, and building graphical user interfaces as well as AI knowledge representations have been successfully demonstrated (Rumbaugh et al., 1991). The foregoing trend in broad application of object-oriented technology indicates that the generality of object-oriented concepts makes it possible to significantly integrate the disciplines of database management, programming languages, software engineering, artificial intelligence, management, and economic sciences. The purpose of this section is not to produce a rigorous description of the object-oriented approach, but rather to provide some basic concepts that are useful for economists who have substantial experience with AGE modeling but who are new to the field or object-oriented programming and knowledge-based systems. It will focus on showing how object-oriented concepts can represent AGE models and capture the semantics of the modeling process.

2. Core object-oriented concepts

The object-oriented approach is a conceptual process independent of programming language until the final stage. It is a way of thinking abstractly to understand complex systems by using application domain concepts. The basic concepts in the object-oriented approach are objects and classes, message passing, class and component hierarchy, and inheritance. Using these concepts, the object-oriented approach encapsulates the principles of abstraction, encapsulation, modularity, and hierarchy to master complexity.

In object-oriented systems all real world entities are treated as objects. Each object has a state, which is the set of particular values of the attributes defined for the objects, and behavior, which is represented by methods associated with the objects for accessing and manipulating the state (a set of programs with a well-defined interface for their invocation), and a system-wide unique identifier. The state and behavior encapsulated in an object can be accessed only by messages sent to that object. The object itself then selects a suitable method to execute the message received. These messages constitute the public interface of an object. Groups of objects with common attributes and behavior are referred to as a class. Objects that belong to a class are called instances of that class. Both classes and instances are objects. An instance of a class can be created by sending a message to that class that has the method for creating objects. Associated with each attribute of a class is a domain defining the values it may take. The domain may be a primitive or composite class. A primitive class is one that has only instances. A composite class has attributes whose domain is also some classes. This type of class attributes can be used to capture aggregate properties of the instances that belong to a given class.

---

2The discussion of core object-oriented concepts are based on Chapter 2 of Kim (1990).
Class hierarchy refers to the relationship between superclass and subclasses. It captures the generalization/specialization relationship (Is-A) between one class and a set of classes specialized from it. The superclass contains properties (both attributes and methods) common to all subclasses, and each subclass has some specific properties of its own. Inheritance is a reusability mechanism through class hierarchy such that a subclass shares properties from all its superclasses. An object accumulates properties from each level of class hierarchy. This 'a kind of' type relationship corresponds to one of the most fundamental aspects of human thinking: the ability to perceive similarities and differences among objects, and to thereby classify them.

Component hierarchy represents the aggregation relationship (part-whole) among objects. It provides a way to decompose the world into structured modules that can be easily understood. Thus 'a part of' type relationship differs from class hierarchy by not allowing inferences of properties.

Abstraction is one of the fundamental ways that humans cope with complexity. It is the process of capturing the essential characteristic of an object. Encapsulation – also known as information hiding, on the other hand, helps manage the complexity by hiding all the details of an object that does not contribute to its essential characteristics. Modularity provides a way to cluster logically related objects. Finally, by identifying the ordering or hierarchy of objects, a complicated problem can be greatly simplified.

There are great benefits from applying the object-oriented approach to formulate AGE models. The object-oriented concepts appeal to the working of human cognition since it is easier for many people who have no idea of mathematical models and how a computer works to view the real world in terms of objects and relationship among objects. The object-oriented approach views the real world as a structured collection of objects that collaborate to perform some higher-level behavior. This is consistent with how economists represent an economy in terms of the interaction among a set of autonomous agents who follow certain institutional rules to conduct economic activities such as production, consumption, and trade. Equilibrium of an economy is usually defined as an object of array (prices and quantities) in standard economic textbooks.

In the object-oriented framework, the different representations of AGE models described earlier reflect views at different levels of abstraction/encapsulation about the same economy under study. Constructing the AGE modeling system under a unified object-oriented framework will not only permit the modeling system to take full advantage of the expressive power of object-oriented programming language and software system design technology, but also help the policy makers, modeling experts, and computer programmers to express abstract concepts clearly and to communicate them to each other. It can serve as a consistent methodology for problem definition, model specification, data preparation, interfacing, and programming, thus facilitating translation among the three model representations.
2.2. An object-oriented view of AGE models

AGE models have many elements in common. They usually consist of five components (Ginsburgh and Robinson, 1984). The first is a set of economic agents such as firms, households, and government whose behavior is to be analyzed. Each agent has a set of endowments that can be used as production factors, such as labor and capital, and an economic account that records his revenues and expenditures. The second component consists of behavioral rules for these agents that reflect their assumed motivation such as profit maximization for firms and utility maximization for consumers. The third is a set of signals observed by these agents on which they make their economic decisions, such as market prices or government rationing quotas. The fourth is the institutional structure of the model economy, which are the rules of the game by which various agents interact. For example, perfect competition implies that each agent is a price taker and prices are flexible. And finally, a set of explicit definitions of equilibrium conditions which are 'system constraints' that must be satisfied for the whole economy but which are not taken into account by each individual agent in making his decisions. An equilibrium can be defined as a set of signals such that the resulting decisions of all agents jointly satisfy the system constraints. The signals represent the equilibrating variables of the model. For example, in a perfectly competitive AGE model the assumption that excess demand equals zero in all markets is a system constraint that defines the nature of equilibrium.

One way to represent the above AGE model structure, using object-oriented concepts, would be as follows. First, each type of agent (consumers, firms, factors, and government) can be represented by a composite class. For example, consumers can be represented by a composite class having an attribute corresponding to different utility functions that reflect his/her behavior rule, plus an endowment and a consumption quantity attribute whose domains are a vector of real numbers. Other types of agents can be represented similarly. All those agent type classes are subclasses of a higher-level composite class called accounts with an income and expenditure transaction attribute whose domains are real numbers. Some accounts also have physical flow attributes. Second, the signals and system constraints can also be represented as composite classes. For example, in a perfectly competitive AGE model, price is the only signal. Different types of prices can be represented as a composite class with an attribute of cardinality and a level whose domains are integers and vectors of real numbers, respectively. Third, the social accounting matrix underlying the AGE model is also a composite class with a primitive attribute of time and attributes of row and column made up by corresponding income and expenditure transaction

---

Footnote: For the entire model, these accounts can be summarized in a social accounting matrix (SAM), whose rows and columns give the income and expenditure flows for each of the agents.
attributes for all accounts. Finally, the AGE model itself is then represented by a composite class, which is an aggregation of class accounts, signals, system constraints, and an accounting table with a set of model solution procedures as its methods, plus simple attributes such as name whose domain is the primitive class 'string' and the dimension of accounts whose domain is integer. An instance of this meta-class is a special AGE model.

Given the above interpretation of an AGE model as a collection of objects with class hierarchy, the model formulation, manipulation, and solution can be interpreted as sending messages to those objects. In most object-oriented models, messages are arranged in standardized groups known as Protocols. For example, every class must have a Protocol for creating, modifying, or deleting its instance, its place in the class hierarchy, and its method-dictionary (Lenard, 1993). The model solution procedure can be invoked by sending a message such as 'solving' to the composite class, AGE models.

3. A frame-based, object-oriented representation of AGE models for world trade

A central issue in the formal representation of descriptive knowledge in a knowledge-based system is the choice of a knowledge representation scheme (KRS). The complexity of AGE models led us to adopt an object-oriented framework widely used in Artificial Intelligence based on class frames (Minsky, 1975). In this scheme knowledge is partitioned into modules called frames, which are data structures that include all the relevant information about an object of the problem domain.

There are two basic elements in frames: slots and facets. In general, a slot provides information describing the pertinent details of an object that the frame is intended to represent. A slot contains one or more facets, which can be filled by numerical values, default, range, procedures, demons (called when needed), rules, descriptive or graphical display information, functional forms, economic assumptions, and any type information, even another frame or frame relationship. Thus computational procedures can be directly attached to frames and other KRS such as production rule modules can be easily incorporated. This gives a frame-based system great power and flexibility of representation. It provides a natural means for organizing different forms of information into structures. It first clusters knowledge into individual frames, then logically links them through generalization, aggregation, and association relationships, which are implemented by slots called Is-A and Is-Part-Of that store the name of the next genetic objects the given objects are related to.

Desirable features of frames in representing AGE models include the distinctions between classes and instances of classes through the use of class frame and member frames, easy to understand by policy makers, graphical interface capability, and the hierarchy of frames that reflect the level of abstraction. In
such representations AGE models are a structured collection of communicating objects such as economic agents represented by frames. Since most people have difficulty in directly conceptualizing a problem in terms of its mathematical formulations, structuring the model in real economy objects that are much closer to user’s views, allowing the user to state his/her problem in non-mathematical terms, making it much easier for a frame-based system to use domain semantics and serve as a means for the translation from user’s view of the problem to its mathematical formulation. In addition, each frame can have an internal state and an external graphical representation for the interface. This provides the means for building a graphical interface with object graphics.

These object graphics as frame ‘windows’ are a convenient means for the user to visualize and interact directly with the model components. This capability is a natural outcome of following the object-oriented approach because it also provides the technology for encoding image-based, user-interface constructs such as interactive editors/browsers, multiple windows, and pop-up menus which are essential for a truly user-oriented modeling environment (Pracht, 1990). Moreover, the hierarchy of frames reflects the level of model abstraction/encapsulation. It enables the model developer to work from a higher level of abstraction to progressively lower levels. At any given level, all details of lower levels are hidden. Each lower level in the hierarchy contains more primitive objects. This object-oriented, frame-based approach provides a convenient basis for direct user manipulation models and their components through frames and frame linkages. A frame representation of major objects in AGE models for world trade corresponding to its mathematical representation given in the Appendix is listed below:

<table>
<thead>
<tr>
<th>Frame: AGE MODELS</th>
<th>Frame: REGIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master frame: AGE MODELS</td>
<td>Is-part-of: AGE MODELS</td>
</tr>
<tr>
<td>Type: Composite class</td>
<td></td>
</tr>
<tr>
<td>Member slot: name and interpretation</td>
<td>Member slots: accounts</td>
</tr>
<tr>
<td>Type: Primitive class</td>
<td>Type: abstract class</td>
</tr>
<tr>
<td>Domain: string</td>
<td>Domain: factors, producers, household, government,</td>
</tr>
<tr>
<td></td>
<td>investor, world, shipping</td>
</tr>
<tr>
<td></td>
<td>Set definition: $N \in {F, I, H, G, V, W, S}$</td>
</tr>
<tr>
<td></td>
<td>Index: $[n, r]$</td>
</tr>
<tr>
<td></td>
<td>Cardinality: $N - n_1 + n_2 + n_3 + 4$</td>
</tr>
<tr>
<td>Member slot: assumptions</td>
<td>Member slot: system constraints</td>
</tr>
<tr>
<td>Type: composite class</td>
<td>Type: abstract class</td>
</tr>
<tr>
<td>Domain: system defined set</td>
<td>Domain: factors, products, model closure</td>
</tr>
<tr>
<td></td>
<td>Cardinality: $n_1 + n_2 + 1$</td>
</tr>
<tr>
<td></td>
<td>Member slot: SAM tables (table T)</td>
</tr>
<tr>
<td></td>
<td>Type: composite class</td>
</tr>
<tr>
<td></td>
<td>Dimension: $N^2$</td>
</tr>
<tr>
<td></td>
<td>Method: collect all account’s transactions</td>
</tr>
<tr>
<td>Member slot: regions</td>
<td></td>
</tr>
<tr>
<td>Type: composite class</td>
<td></td>
</tr>
<tr>
<td>Domain: user defined</td>
<td></td>
</tr>
<tr>
<td>Index: $[s, r]$</td>
<td></td>
</tr>
<tr>
<td>Method slot: solution procedures</td>
<td></td>
</tr>
<tr>
<td>Type: solver</td>
<td></td>
</tr>
<tr>
<td>Domain: system defined</td>
<td></td>
</tr>
</tbody>
</table>


Frame: ACCOUNTS
Is-part-of: REGIONS

Member slot: name and interpretation
Type: primitive class
Domain: string

Member slot: account type
Type: primitive class
Domain: agent, activity

Member slot: transactions
Type: abstract class
Domain: incomes, payments
Notation: \( t_{nk} \in T_r, \ n, k \in N \)

Member slot: budget balance
Method: sum of incomes equal sum of payments for each account
Index: \([n]\)

Member slot: behavior rules
Demon: if agents
Type: composite class
Domain: functional forms
Method: user selected

Member slot: quantities
Type: abstract class
Domain: consumption, production, invest demand, inter demand, factor demand, domestic sales, export, import, trade flows, endowment, shipping demand, shipping supply
Set definition: \( D \in \{Q, D, IX, DF, DX, EX, MX, X, FS\} \)

Member slot: prices
Type: abstract class
Domain: f.o.b, c.i.f., output, domestic, export, demand, import, factor, invest, shipping
Set definition: \( P \in \{PW, PC, P, PD, PE, PX, PM, PF, PK, PRT\} \)

Frame: HOUSEHOLDS
Superclass (Is-a): ACCOUNTS-AGENTS
Type: composite class
Domain: user defined
Cardinality: \( n_3 \)

Member slot: endowments \((FS)\)
Type: composite class, fixed
Associate price: factor \((PF)\)
Cardinality: \( n_1 \times n_3 \)
Notation: \( FS_{kj} \forall h \in H, \forall f \in F \)
Index: \([h, f, r]\)
Method: user input

Member slot: consumption \((C)\)
Type: composite class
Associate price: demand \((PX)\)
Cardinality: \( n_1 \times n_3 \)
Notation: \( C_{hj} \forall h \in H, \forall i \in I, \forall r \in R \)
Index: \([h, i, r]\)
Method: system determined

Member slot: saving \((HSAV)\)
Type: primitive class
Domain: real number
Index: \([h, r]\)

Frame: PRODUCERS
Superclass (Is-a): ACCOUNTS-AGENTS
Type: composite class
Domain: user defined
Cardinality: \( n_2 \)

Member slot: production \((Q)\)
Type: composite class
Associate price: output \((P)\)
Cardinality: \( n_2 \)
Notation: \( Q_{ijr} \forall i \in I, \forall r \in R \)
Index: \([i, r]\)
Method: system determined

Member slot: factor demand \((DF)\)
Type: composite class
Associate price: factor \((PF)\)
Cardinality: \( n_1 \times n_3 \)
Notation: \( DF_{ijr} \forall f \in F, \forall i \in I, \forall r \in R \)
Index: \([i, f, r]\)
Method: system determined

Member slot: I/O coefficient
Type: primitive class
Cardinality: \( n_2^2 \)
Index: \([i, i, r]\)
Notation: \( io_{ijr} \forall i, j \in I, \forall r \in R \)
Method: user input
Member slot: inter demand (IX)
Type: composite class
Associate price: demand (PX)
Notation: $\sum_{i \in I} X_{ir}, \forall i \in I, \forall r \in R$
Index: $[i, r]$
Method: system determined

Frame: FACTORS
Superclass (Is-a): ACCOUNTS ACTIVITY
Type: composite class
Domain: default, user defined
Cardinality: $n_1$

Member slot: factor demand
Type: Composite class
Associate price: factor (PF)
Cardinality: $n_1$
Notation: $DF_{jr} = \sum_{j \in J} DF_{ijr}, \forall f \in F, \forall r \in R$
Index: $[j, r]$
Method: sum over all firms

Member slot: endowment
Type: Composite class
Associate price: factor (PF)
Cardinality: $n_1$
Notation: $FS_{fr} = \sum_{h \in H} FS_{fh}, \forall f \in F, \forall r \in R$
Index: $[j, r]$
Method: sum over all consumers

Frame: GOVERNMENT
Is-part-of: ACCOUNTS AGENT

Member slot: endowments (GFKS)
Type: primitive class, fixed
Associate price: factor (PF)
Domain: real number
Index: $[k, r]$
Method: user input

Member slot: consumption (GC)
Type: composite class
Associate price: demand (PX)
Cardinality: $n_2$
Notation: $GC_{ir}, \forall i \in I, \forall r \in R$
Index: $[i, r]$
Method: system determined or user input

Member slot: saving (GSAV)
Type: primitive class
Domain: real number
Index: $[r]$
Method: system determined or user input

Frame: SHIPPING
Is-part-of: ACCOUNTS ACTIVITY

Member slot: tax rate
Type: abstract class
Domain: indirect tax, tariff, export subsidy
(INDTAX, TARIFF, ETAX)
Method: user input

Frame: INVESTOR
Is-part-of: ACCOUNTS ACTIVITY

Member slot: invest demand
Type: composite class
Associate price: demand (PX)
Cardinality: $n_2$
Notation: $ID_{ir}, i \in I, \forall r \in R$
Index: $[i, r]$
Method: system determined

Member slot: income (INV)
Index: $[r]$
Methods: sum of all type savings, system determined or user defined

Frame: WORLD
Is-part-of: ACCOUNTS ACTIVITY

Member slot: export (EX)
Type: composite class
Associate price: export (PE)
Cardinality: $n_2$
Notation: $EX_{ir}, i \in I, \forall r \in R$
Index: $[i, r]$
Method: system determined

Member slot: import (MX)
Type: composite class
Associate price: export (PM)
Cardinality: $n_2$
Notation: $MX_{ir}, i \in I, \forall r \in R$
Index: $[i, r]$
Method: system determined

Member slot: balance of trade (BOT)
Type: primitive class
Domain: real numbers
Index: $[r]$
Method: system determined or user input
Associate price: f.o.b or c.i.f. (PW or PC)
Cardinality: $n_z \times R^2$
Notation: $X_{ip}, i \in I, \forall s, r \in R$
Index: $[i, s, r]$
Method: system determined

Member slot: margin
Type: primitive class
Cardinality: $n_z \times R^2$
Notation: $tr_{isr}, i \in I, \forall s, r \in R$
Index: $[i, s, r]$
Method: user input

Member slot: shipping demand ($TRQD$)
Type: composite class
Associate price: shipping ($PRT$)
Cardinality: $n_z$
Notation: $TRQDi, i \in I, \forall s, r \in R$
Index: $[i, r]$
Method: system determined

Member slot: shipping supply ($TRQS$)
Type: primitive class
Associate price: output ($P$)
Index: $[s, r]$
Method: system determined

Frame: QUANTITIES
Is-part-of: ACCOUNTS

Member slot: name
Type: composite class
Domain: string

Member slot: quantity type
Type: primitive class
Domain: demand, supply

Member slot: amount
Type: primitive class
Domain: vector of real numbers

Frame: ENDOWMENT ($FS$)
Superclass (Is-a): QUANTITIES SUPPLY
Is-part-of: CONSUMERS, FACTORS, GOVERNMENT
Index: $[h, f, r]$

Frame: CONSUMPTION ($C$)
Superclass (Is-a): QUANTITIES DEMAND
Is-part-of: CONSUMERS, PRODUCT MARKET
Index: $[h, i, r]$

Frame: PRODUCTION ($Q$)
Superclass (Is-a): QUANTITIES
Is-part-of: PRODUCERS, PRODUCT MARKET
Index: $[i, r]$

Frame: FACTOR DEMAND ($DF$)
Superclass (Is-a): QUANTITIES DEMAND
Is-part-of: PRODUCERS, FACTORS
Index: $[i, f, r]$

Frame: INTER DEMAND ($IX$)
Superclass (Is-a): QUANTITIES DEMAND
Is-part-of: PRODUCERS
Index: $[i, j, r]$

Frame: INVEST DEMAND ($ID$)
Superclass (Is-a): QUANTITIES DEMAND
Is-part-of: INVESTOR
Index: $[i, r]$

Frame: EXPORT ($EX$)
Superclass (Is-a): QUANTITIES DEMAND
Is-part-of: WORLD
Index: $[i, r]$

Frame: DOMESTIC SALES
Superclass (Is-a): QUANTITIES SUPPLY
Is-part-of: PRODUCT MARKET
Index: $[i, r]$

Frame: IMPORT ($MX$)
Superclass (Is-a): QUANTITIES SUPPLY
Is-part-of: WORLD
Index: $[i, r]$

Frame: TRADE FLOWS ($X$)
Superclass (Is-a): QUANTITIES
Is-part-of: SHIPPING
Index: $[i, s, r]$

Frame: SHIPPING DEMAND ($TRQD$)
Superclass (Is-a): QUANTITIES
Is-part-of: SHIPPING
Index: $[i, r]$

Frame: SHIPPING SUPPLY ($TRQS$)
Superclass (Is-a): QUANTITIES
Is-part-of: SHIPPING
Index: $[r]$
Frame: PRICES
Is-part-of: ACCOUNTS

Member slot: name and interpretation
Type: primitive class
Domain: string

Member slot: levels
Type: primitive class
Domain: real number
Method: system determined

Frame: f.o.b. PRICE (PW)
Superclass (Is-a): PRICE
Index: [i, s, r]

Frame: c.i.f. PRICE (PC)
Superclass (Is-a): PRICE
Index: [i, s, r]

Frame: OUTPUT PRICE (P)
Superclass (Is-a): PRICE
Index: [i, r]

Frame: FACTOR PRICE (PF)
Superclass (Is-a): PRICE
Index: [i, r]

Frame: DOMESTIC PRICE (PD)
Superclass (Is-a): PRICE
Index: [i, r]

Frame: EXPORT PRICE (PE)
Superclass (Is-a): PRICE
Index: [i, r]

Frame: DEMAND PRICE (PX)
Superclass (Is-a): PRICE
Index: [i, r]

Frame: IMPORT PRICE (PE)
Superclass (Is-a): PRICE
Index: [i, r]

Frame: INVEST PRICE (PK)
Superclass (Is-a): PRICE
Index: [r]

Frame: SHIPPING PRICE (PRT)
Superclass (Is-a): PRICE

Frame: SYSTEM CONSTRAINTS
Is-part-of: REGIONS

Member slot: name and interpretation
Type: primitive class
Domain: string

Member slot: demand side
Type: composite class
Domain: factors, producers, shipping, fund

Member slot: supply side
Type: composite class
Domain: factors, producers, shipping, fund

Method slot: demand side equals supply side

Frame: FACTOR MARKETS
Superclass (Is-a): SYSTEM CONSTRAINTS

Member slot: name and interpretation
Type: primitive class
Domain: string
Index: [f, r]

Member slot: total supply (supply side)
Method: sum over endowments of all ACCOUNT AGENT

Member slot: total demand (demand side)
Method: sum of factor demand of all PRODUCERS

Frame: PRODUCT MARKETS
Superclass (Is-a): SYSTEM CONSTRAINTS

Member slot: name and interpretation
Type: primitive class
Domain: string
Index: [i, r]

Member slot: total demand (demand side)
Method: system deduced, sum over QUANTITY DEMAND
Notation: \( TX_r = \sum_{h \in H} C_{afr} + GC_{ir} + 1D_r + IX_r \), \( \forall i \in I, \forall r \in R \)

Member slot: total supply
Method: system deduced, sum over QUANTITY SUPPLY
Notation: \( DX_r + MX_r \), \( \forall i \in I, \forall r \in R \)
Frame: MACRO CLOSURE (fund market)
Superclass (Is-a): SYSTEM
CONSTRAINTS

Member slot: relation between saving and investment at aggregate level
Index: [r]
Method: user defined

Member slot: numeraire price or index
Method: user defined
Index: [r]

Member slot: gross investment (demand side) (INV)
Type: primitive class
Domain: real numbers
Method: user defined

Member slot: total saving (supply side)
Method: sum of all type saving
Notation: \( \sum SAV_\alpha + GSAV_r - BOT_r \)

Frame: BUDGET BALANCE
Is-part-of: ACCOUNTS

Member slot: income (I)
Type: primitive class
Domain: real number
Method: sum of income amount
Notation: \( I_i = \sum_{n=1}^{N_i} I_{ni} \)

Member slot: expenditure (E)
Type: primitive class
Domain: real number
Method: sum of expenditure amount
Notation: \( E_i = \sum_{k=1}^{N_k} I_{nk} \)

Frame: SHIPPING MARKET
Superclass (Is-a): SYSTEM CONSTRAINTS

Member slot: total shipping supply
Method: sum of shipping supply of all region
Notation: \( \sum_{R} TRQS_R \)

Member slot: total shipping demand
Method: sum of shipping demand of all region and all products
Notation: \( \sum_{R} \sum_{\alpha} TRQD_{\alpha R} \)

Frame: FUNCTIONAL FORM
Is-part-of: BEHAVIOR RULES

Member slot: name
Type: primitive class
Domain: string

Member slot: functional form
Type: composite class

Frame: INCOMES (I)
Superclass (Is-a): TRANSACTIONS
Index: n over N

Frame: EXPENDITURES (E)
Superclass (Is-a): TRANSACTIONS
Index: n over N

Frame: TAX RATE
Is-part-of: GOVERNMENT
4. Toward an object-oriented, knowledge-based AGE modeling system

The above frame-based representation of objects in AGE models can be implemented in an object-oriented, knowledge-based modelling system as the core structure of its model base. Such a system applies knowledge-based techniques to dynamically construct an AGE model corresponding to a semantic model structure provided by end-users (policy makers). It maintains and stores algebraic structures of equations as canonical objects and manipulates them to generate an AGE model specification that can be represented in a conventional programming language. In such a system, model construction begins with a semantic structure specification. The details of how this specification may be obtained is the task of user interface design. It is declarative and oriented towards capturing qualitative aspects of the policy problem under study. This includes the identification of the important objects and their linkages, such as the specification of model dimension, choice of agents and their behavior rules, selection of functional form and macro-closure, and so on. While equations that correspond to the semantic structure are not supplied by the user, it is the responsibility of the system to construct the algebraic equations and generate a computable specification of the model in a target language that supports mathematical modeling. This is different from most current modeling systems which require users to construct and supply a computer-recognized specification of the algebraic form of the model.

There are two stages involved in the transformation of the semantic structure specification to a computable algebraic specification. The first stage involves transforming the user-defined semantic representation of the model into different types of canonical objects, which represent knowledge about the algebraic structure of the model and implemented by frames. Once the types of these objects are obtained, equation construction rules are employed to build the equation form of the model. The algebraic model then is in turn transformed into the syntax of a target modeling language, while this part of transformation
Fig. 1. Relations among major frames in a world trade model.
is purely syntactic. Therefore, three types of knowledge are involved in such a model construction process: domain-specific knowledge, modeling knowledge, and syntactic knowledge of the target modeling language used to represent computer-executable specifications.

To some extent the architecture to implement such a modeling system is similar to those used in automatic programming systems (Barstow, 1977). It may consist of six components: (1) a modeling language subsystem that provides support for model algebraic representation and numerical solution; (2) a database management subsystem that performs the tasks of data validation, output report writing, sensitivity, and 'what if' analysis; (3) an econometric software that conducts parameter estimations and statistical analysis when they are needed; (4) a model base that stores the core structure of the AGE model as canonical objects and reusable models; (5) a knowledge base including a GAMS code generator, an equation builder that contain rules on how to manipulate the objects in the model base to dynamically construct or revise a model instance for a specific policy objective based on semantic specification from end-users, and a rule base that guides simulation results analysis; and (6) a graphical, window-driven interface that integrates all the five components and makes model specification, simulation scenarios design, and result analysis easier to conduct by the policy maker with little computer and mathematical knowledge. The architecture of such a system is depicted in Fig. 2.

This is a loosely coupled system. Communication among its components is via file passing or by a common database (blackboard strategy). It employs a domain-specific window-driven problem specification interface, multiple knowledge bases for dynamic model construction, and GAMS as the target computer language. Most of its components can take advantage of the existing software products in the market. Besides using GAMS as its modeling language, FOCUS or Fox Pro can be adopted as its database management system, SHAZAM can be selected as its econometric program.

The modeling expert subsystem and the graphical window-driven integrated interface, as well as the model base that is organized by class frames, can be implemented in LEVEL 5 OBJECTS (Information Builders Inc., 1993), a total object-oriented programming environment. It has system classes which provide a rich array of built-in logic and object tools, direct access and complete joining facility with FOCUS, DBASE, Lotus, and ASCII files, a Graphical User Interface (windows) development editor, hybrid knowledge representation (support class frames), and multiple inference strategies. This means the technologies for implementing such a modeling system have already matured (of course, the system can be more flexible and integrated by implementing it in a high-level object-oriented language such as C++). The major challenge now is the development of its model base and knowledge base,
Fig. 2. A loosely coupled knowledge-based CGE modelling system.
which need joint effort from economists and computer scientists. The frame representation of objects in AGE models presented in this paper is an initial step in this direction.

A major feature of the proposed modeling system is that it uses a knowledge-based approach to aid the dynamic construction of AGE models applied to policy analysis. By adding an 'intelligent' model formulation and results interpretation component, it will greatly facilitate the use of AGE models by users unfamiliar with mathematical models and computer programming. In addition, the system will be developed under a unified object-oriented framework that facilitates frame-based AGE model representation. This provides the capacity to apply domain-independent model building rules instead of instantly storing the model in equation forms, thus adding great flexibility in revising a model after it has been generated. This intelligent modeling system will accumulate and store all the relevant knowledge used to construct and generate executable specifications of AGE models in its knowledge base. Using the inference engine, and through the equation builder and the automatic GAMS code generator, the system will conduct algebraic model specification, model components assembling, policy instruments selections, simulation scenarios design, and policy recommendations just like an experienced AGE model developer.

5. Concluding remarks

The discussion in this paper represents an initial step of an ambitious research effort to apply the object-oriented approach to economic system modeling. The intention is to illustrate the usefulness of the object-oriented framework to the AGE modeling community. History shows that the advance of applied general equilibrium modelling needs the concurrent development of economic theory, modeling methods, and computer support technology. Although complete implementation of the proposed modeling system is not in the scope of this paper, the object-oriented approach outlined here provides an ideal framework for all the three trends to interact. The underlying object-oriented concepts are also the common thread linking frame-based knowledge representation and reasoning systems, object-oriented programming environments, and object-oriented user interfaces. Therefore, they are the key to building intelligent computer-aided economic modeling systems. In such a system, the model developer and policy maker will work with familiar objects with prescribed behavior from the real world, constructing AGE models will appear as an intuitive rather than abstract process. It needs more understanding about the substance of economics and the logic behind the problem at hand than mathematical notation.
Appendix

Algebraic representation of the AGE world trade model

This appendix provides a detailed description of the algebraic structure of the world trade model. The complete set of core equations describing the model is given in Table A.1. Definition of variables and parameters is given in Tables A.2 and A.3, respectively.

A.1. Notation

(i) Region name: CH (China), HK (Hong Kong), TW (Taiwan), JP (Japan), US (United States), EC (12 members of European Community), ROW (rest of the world).

(ii) Industry name: AF (food and agriculture), ME (mineral and energy), BI (basic manufactured intermediates), MT (machinery and transport equipment), OMT (other manufactures), SV (services).

(iii) Factor name: LD (agriculture land), ULB (unskilled labor), SLB (skilled labor), K (capital).

(iv) Subscripts and set definition:
- Regions are defined in set $R$ and indexed by $r$ or $s$, $r, s \in R = \{CH, HK, TW, JP, US, EC, ROW\}$
- Industries are defined in set $I$ and indexed by $i$ or $j$, $i, j \in I = \{AF, ME, BI, MT, OMT, SV\}$
- Primary factors are defined in set $F$ and indexed by $f$, $f \in F = \{LD, ULB, SLB, K\}$.

(v) Conventions: Uppercase English letter indicate variables, unless they have a bar on top, in which case that variable always set exogenously. Greek letter or lowercase English letter refer to parameters, which need to be calibrated or supplied from exogenous sources. The first index of variable or parameter represents the region or sector supplying goods; the next one represents the region or sector purchasing goods.

A.2. Equation description

Eqs. (1) and (2) define the relation between border (world) prices and internal prices, while Eqs. (3)–(5) define price indices for composite imported goods, composite demand goods, and the firm's composite output, respectively. In Eqs. (3) and (4) the price indices are the unit cost functions, while in Eq. (5) they are unit revenue functions, all of which are dual to the corresponding unit quantity aggregator functions. Since CES functions are used as the building blocs of the basic model, and this quantity aggregator function is homogeneous of degree one, the total costs can be written as total quantity multiplied by unit
cost (Varian, 1984, p. 28). This implies that the average cost, under cost minimization, is independent of the number of units produced or purchased. Thus the unit cost function also stands for the price of aggregated commodity. Eq. (6) defines the unit price for investment goods, which is the sum of all the value of its contents. Eq. (7) gives the value-added or net price per unit output. It follows from the unit revenue function, Eq. (4), and the total production cost function (fixed cost assumed to be zero in constant returns case), given in Eq. (10) and will be discussed below. Eq. (8) defines the numeraire in the model.

Eq. (9) defines the total factor cost, which equals unit factor cost with a functional form similar to that of Eqs. (3)–(5), multiplied by the quantities of total output. Total cost function \( TC_i \) in Eq. (10) is composed of factor cost and intermediate cost, plus indirect taxes. It is the result of cost minimization by the representative firm in each sector with respect to its factor and intermediate inputs, subject to the Leontief production function. The strong separability assumption in technology guarantees that the cost function is additively separable in its factor and intermediate cost components. Eq. (11) gives the factor demand functions, which are obtained by taking partial derivatives of the factor cost function, Eq. (9), with respect to the relevant factor price, according to Shephard’s lemma. Eq. (12) describes the intermediate demand for the composite goods of sector \( i \) which results from the fixed proportion input requirement assumption. Eqs. (13)–(16) are the domestic and export supply functions corresponding to the constant elasticity of transformation (CET) function commonly used in today’s CGE models. They are derived from revenue maximization, subject to the \( CET \) function, in a way similar to the derivation of factor demand functions. Eq. (17) aggregates exports by the representative firm in each region, which implies that producers only differentiate output sold in domestic and foreign markets, but do not differentiate exports by destinations.

Eq. (18) is the consumer demand functions, which is the Linear Expenditure System derived from maximizing a Stone–Geary utility function subject to household disposable income, which is given in Eq. (28). Eqs. (19) and (20) give government and investment demands, which are generated from maximization of Cobb–Douglas utility functions subject to their respective budget constraints, Eqs. (29) and (39). Eqs. (21)–(23) are demand functions for domestic goods, for aggregate imported goods, and for imported goods by sources, respectively. They describe the cost-minimizing choice of domestic and import purchases, as

---

4This is a proposition provided in Varian (1984). It states that if the production function exhibits constant returns to scale, the corresponding cost function may be written as \( C(W, Q) = Q \times C(W, 1) \).
well as import sources of commodity \( i \) by region \( r \). They are derived from corresponding cost functions according to Shephard's lemma in a way similar to the derivation of factor demand functions (taking partial derivatives of the cost function with respect to the relevant component prices). Because of the linear homogeneity of the CES function, the cost function that is dual to the commodity aggregator can be represented by its unit cost function, Eqs. (3) and (4), multiplied total quantity demanded.

Eqs. (24) and (25) describe the supply side of international shipping industry. Eq. (25) states that at equilibrium, the returns from shipping activity must cover its cost. Just as other industries in the model, it also earns zero profit. Eq. (25) describes the demand for each region's service sector exports to the international shipping industry, which is generated by the assumed Cobb–Douglas technology in this industry. The next two equations, Eqs. (26) and (27), refer to the demand side of the international shipping industry. The demand for shipping services associated with commodity \( i \) in region \( r \) is generated by a fixed proportion input requirement (Leontif) coefficient \( tr_{isr} \), which is routine/commodity-specific, Eq. (26). In equilibrium, the total demand of shipping service must equal its total supply, Eq. (27).

Eqs. (30)–(32) determine government revenue from indirect taxes, tariffs, and export taxes (its negative equals a subsidy), respectively, while Eqs. (33)–(44) define household saving, government saving, and the balance of trade (foreign saving) in each region.

Eqs. (36)–(39) are system constraints that the model economy must satisfy. For every sector in each region, the supply of the composite goods must equal total demand, Eq. (36), which is the sum of household consumption \( (C_{ir}) \), government purchases \( (GC_{ir}) \), investment \( (ID_{ir}) \), and the firm's intermediate demand \( (rX_{ir}) \). Similarly, the demand for each factor in every region must equal the exogenously fixed supply, Eq. (37). In this dual formulation, output in each region is determined by demand and sectoral equilibrium are determined in Eq. (38), unit output price equals average cost, which is also the zero profit condition. Eq. (39) describes the macroeconomic equilibrium identity in each region, which is also the budget constraint for the investor. Since all agents in each region (households, government, investors, and firms) satisfy their respective budget constraints, it is well-known that the sum of the excess demand for all goods is zero; that is Walras' law holds for each region. Therefore, there is a functional dependence among the equations of the model. One equation is redundant in each region and thus can be dropped.

The last two equations, Eqs. (40) and (41), define GNP. \( GNP_{r} \) is defined from the demand side, which is the sum of the three categories of final demand corrected by the balance of trade. \( GNP_{VA} \) is calculated from the supply side and equals the sum of value-added (including indirect taxes) plus tariffs, less export subsidies. The two GNP can also be used to specify a GNP price deflator as an alternative choice of numeraire for the model.
Table A.1
Core equations of world general equilibrium trade model

**PRICE**

(1) \[ PW_{ir} = (1 + te_{ir}) \times \left( \frac{1}{ER_r} \right) \times PE_{ir} \]

(2) \[ PC_{ir} = (1 + trs_{ir}) \times PW_{ir} \]

(3) \[ PM_{ir} = \frac{1}{h_{ir}} \times \left( \sum_{s \in R} \delta_{ir}^{s} \times \left[ (1 + tm_{ir}) \times ER_s \times PC_{ir} \times (1 - \sigma_{ir}) \right] \right) \]

(4) \[ PX_{ir} = \frac{1}{f_{ir}} \times \left( \sum_{s \in R} \delta_{ir}^{s} \times PD_{ir}^{1 - \sigma_{ir}} + (1 - \sigma_{ir}) \times PM_{ir}^{1 - \sigma_{ir}} \right) \times \frac{1}{1 - \sigma_{ir}} \]

(5) \[ P_{ir} = \frac{1}{\lambda_{ir}} \times \left( \kappa_{ir}^{s} \times PD_{ir}^{1 - \sigma_{ir}} \right) + (1 - \kappa_{ir}^{s}) \times PE_{ir}^{1 - \sigma_{ir}} \]

(6) \[ PK_{r} = \sum_{i \in I} kio_{ir} \times PX_{ir} \]

(7) \[ PV_{ir} = P_{ir} - \sum_{j \in I} io_{ir} \times PX_{ir} \]

(8) \[ PID_{r} = \sum_{i \in I} pwt_{ir} \times PD_{ir} \]

**PRODUCTION**

(9) \[ VC_{ir} = \frac{Q_{ir}}{A_{ir}} \times \left( \sum_{s \in R} \delta_{ir}^{s} \times PE_{ir}^{1 - \sigma_{ir}} \right) \times \frac{1}{1 - \sigma_{ir}} \]

(10) \[ TC_{ir} = VC_{ir} + \sum_{j \in I} i_{ir} \times PX_{ir} \times Q_{ir} + \sum_{j \in I} i_{ir} \times Q_{ir} \times tind_{ir} \times Q_{ir} \]

(11) \[ DF_{ir} = \left( \frac{Q_{ir}}{A_{ir}} \right)^{1 - \sigma_{ir}} \times \left( \frac{VC_{ir}}{PF_{ir}} \right)^{\sigma_{ir}} \]

(12) \[ IX_{ir} = \sum_{i \in I} i_{ir} \times Q_{ir} \]

(13) \[ DX_{ir} = \left( \frac{1}{\lambda_{ir}} \right)^{1 - \sigma_{ir}} \times \left( \kappa_{ir}^{s} \times \frac{P_{ir}}{PD_{ir}} \right)^{\sigma_{ir}} \times Q_{ir} \] for \( i \neq s \)

(14) \[ DX_{sv,ir} = \left( \frac{1}{\lambda_{sv,ir}} \right)^{1 - \sigma_{sv,ir}} \times \left( \kappa_{sv,ir} \times \frac{P_{sv,ir}}{PD_{sv,ir}} \right)^{\sigma_{sv,ir}} \times (Q_{sv,ir} - TRQS_{sv}) \]

(15) \[ EX_{ir} = \left( \frac{1}{\lambda_{ir}} \right)^{1 - \sigma_{ir}} \times \left( (1 - \kappa_{ir}) \times \frac{P_{ir}}{PE_{ir}} \right)^{\sigma_{ir}} \times Q_{ir} \] for \( i \neq s \)

(16) \[ EX_{sv,ir} = \left( \frac{1}{\lambda_{sv,ir}} \right)^{1 - \sigma_{sv,ir}} \times \left( (1 - \kappa_{sv,ir}) \times \frac{P_{sv,ir}}{PE_{sv,ir}} \right)^{\sigma_{sv,ir}} \times (Q_{sv,ir} - TRQS_{sv}) \]

(17) \[ EX_{ir} = \frac{1}{PE_{ir}} \times \sum_{s \in R} \frac{ER_{s}}{(1 + te_{ir})} \times PW_{irs} \times X_{irs} \]
Table A.1 (continued)

DEMAND

\[ C_{ir} = \gamma_{ir} + \frac{\beta_{ir}}{PX_{ir}} \times \left( HDI_r - SAV_r - \sum_{j \in I} PX_{jr} \times \gamma_{jr} \right) \]

\[ GC_{ir} = \frac{\theta_{ir}}{PX_{ir}} \times GSP_r \]

\[ ID_{ir} = \frac{\kappa_{ir}}{PX_{ir}} \times INV_r \]

\[ DX_{ir} = \left( \frac{1}{\mu_{ir}} \right)^{1 - \alpha_{mi}} \times \left( \alpha_{ir} \times \frac{PX_{ir}}{PD_{ir}} \right)^{\alpha_{mi}} \times TX_{ir} \]

\[ MX_{ir} = \left( \frac{1}{\mu_{ir}} \right)^{1 - \alpha_{mi}} \times \left( \left( 1 - \alpha_{ir} \right) \times \frac{PX_{ir}}{PM_{ir}} \right)^{\alpha_{mi}} \times TX_{ir} \]

\[ X_{ir} = \left( \frac{1}{\mu_{ir}} \right)^{1 - \alpha_{ii}} \times \left( \frac{PM_{ir}}{(1 + t_{m_{ir}}) \times ER_r \times PC_{ir}} \right)^{\alpha_{ii}} \times MX_{ir} \]

INTERNATIONAL SHIPPING

\[ TRQ = \frac{1}{PTR} \times \sum_{r \in R} \frac{P_{sv_r} \times TRQS_r}{ER_r} \]

\[ TRQS_r = \frac{\tau_r \times ER_r}{P_{sv_r}} \times PTR \times TRQ \]

\[ TRQD_{ir} = \frac{1}{PRT} \times \left( \sum_{s \in R} \sum_{i \in I} tr_{si_{ir}} \times PW_{irs} \times X_{irs} \right) \]

\[ TRQ = \sum_{r \in R} \sum_{i \in I} TRQD_{ir} \]

INCOME AND SAVING

\[ HDI_r = \sum_{j \in FC} PF_{jr} \times \bar{FS}_{jr} + PF_{k_r} \times (\bar{FS}_{k_r} - \bar{GFKS}_r) - PK_r \times hdr_r \times (\bar{FS}_{k_r} - \bar{GFKS}_r) \]

\[ GR_r = PF_{k_r} \times \bar{GFKS}_r + IDTAX_r + TARRIF_r + TAXE_r - PK_r \times gdr_r \times \bar{GFKS}_r \]

\[ IDTAX_r = \sum_{i \in I} tind_{ir} \times P_{ir} \times Q_{ir} \]

\[ TARRIF_r = \sum_{s \in R} \sum_{i \in I} tm_{is_{ir}} \times ER_r \times PC_{irs} \times X_{irs} \]

\[ TAXE_r = \sum_{s \in R} \sum_{i \in I} te_{is_{ir}} \times PE_{ir} \times X_{irs} \]

\[ SAV_r = MSP_r \times HDI_r \]

\[ GSAV_r = GR_r - GSP_r \]

\[ BOT_r = \sum_{s \in R} \sum_{i \in I} PW_{irs} \times X_{irs} + \frac{P_{sv_{ir}}}{ER_r} \times TRQS_r - \sum_{s \in R} \sum_{i \in I} PC_{irs} \times X_{irs} \]
Table A.1 (continued)

GENERAL EQUILIBRIUM

(36) \[ TX_{ir} = C_{ir} + GC_{ir} + IX_{ir} + ID_{ir} \]

(37) \[ \sum_{i=1}^{n} DF_{fi}, = \frac{FS_{ir}}{n} \]

(38) \[ P_{ir} = TC_{ir}/Q_{ir} \]

(39) \[ INV_{r} = PK_{r} \times (\text{fs}_{ir} - \text{gfs}_{s}r) + \text{gdr}_{r} \times \text{gfs}_{s}r + SAV_{r} + GSAV_{r} - \text{ER}_{r} \times \text{BOT}_{r} \]

GROSS NATIONAL INCOME

(40) \[ GNPVA_{r} = \sum_{i=1}^{n} PV_{ir} \times Q_{ir} + \text{TARRIF}_{r} + \text{TAXE}_{r} \]

(41) \[ GNPR_{r} = \sum_{i=1}^{n} (C_{ir} + GC_{r} + \text{ID}_{ir}) + \text{ER}_{r} \times \text{BOT}_{r} \]

TOTAL NUMBER OF EQUATIONS: \((14 + F) \times R + (18 + F) \times I \times R + 3 \times I \times R^2 + 2 \) (1934)

Table A.2

<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
<th>No. of variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>PW_{ir}</td>
<td>World f.o.b. price for goods from region s to region r</td>
<td>( I \times R^2 ) (294)</td>
</tr>
<tr>
<td>PC_{ir}</td>
<td>World c.i.f. price for goods from region s to region r</td>
<td>( I \times R^2 ) (294)</td>
</tr>
<tr>
<td>PM_{ir}</td>
<td>Price of aggregate imported goods in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>PX_{ir}</td>
<td>Price of composite goods in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>PD_{ir}</td>
<td>Price of domestic products sold at domestic market in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>PE_{ir}</td>
<td>Price of domestic goods for exports in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>P_{ir}</td>
<td>Average output price in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>PF_{ir}</td>
<td>Factor price in region r</td>
<td>( F \times R ) (28)</td>
</tr>
<tr>
<td>PV_{ir}</td>
<td>Price of value added in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>PK_{r}</td>
<td>Price of capital goods in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>ER_{r}</td>
<td>Exchange rate of region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>PID_{r}</td>
<td>Price index in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>Q_{ir}</td>
<td>Sector output in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>VC_{ir}</td>
<td>Variable sector production cost in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>TC_{ir}</td>
<td>Total cost of sector production</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>DF_{ir}</td>
<td>Sector factor demand in region r</td>
<td>( F \times I \times R ) (168)</td>
</tr>
<tr>
<td>IX_{ir}</td>
<td>Sector intermediate demand in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>DX_{ir}</td>
<td>Sector domestic sales in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>EX_{ir}</td>
<td>Domestic goods for exports</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>C_{ir}</td>
<td>Household consumption in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>GC_{ir}</td>
<td>Government spending in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>ID_{ir}</td>
<td>Investment demand in region r</td>
<td>( I \times R ) (42)</td>
</tr>
</tbody>
</table>
Table A.2 (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
<th>No. of variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>TX&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Composite goods demand (supply) in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>MX&lt;sub&gt;isz&lt;/sub&gt;</td>
<td>Sector composite goods imports in region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>X&lt;sub&gt;isz&lt;/sub&gt;</td>
<td>Trade flows from region s to region r</td>
<td>( I \times R^2 ) (294)</td>
</tr>
<tr>
<td>TRQ</td>
<td>Total international transportation supply</td>
<td>1 (1)</td>
</tr>
<tr>
<td>PTR</td>
<td>Price of international shipping service</td>
<td>1 (1)</td>
</tr>
<tr>
<td>TRQDi&lt;sub&gt;r&lt;/sub&gt;</td>
<td>International shipping demand by region r</td>
<td>( I \times R ) (42)</td>
</tr>
<tr>
<td>TRQS&lt;sub&gt;r&lt;/sub&gt;</td>
<td>International shipping service supply by region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>HDI&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Household disposable income in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>MPS&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Marginal propensity to saving of region r's private household</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>GR&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Total government revenue in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>GSP&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Total government spending in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>TARRIF&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Total tariff revenue in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>TAXB&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Total export tax revenue (subsidy expenditure) in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>IDTAX&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Total indirect tax revenue in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>SAV&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Household savings in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>GSAV&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Government saving (deficit) in region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>BOT&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Balance of trade in region r (net capital inflow)</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>INV&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Gross investment by region r</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>GNPVA&lt;sub&gt;r&lt;/sub&gt;</td>
<td>GNP at market price of region r from supply side</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>GNPR&lt;sub&gt;r&lt;/sub&gt;</td>
<td>GNP of region r at base-year price from demand side</td>
<td>( R ) (7)</td>
</tr>
<tr>
<td>FS&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Factor endowment by region r</td>
<td>( F \times R ) (28)</td>
</tr>
<tr>
<td>GFKS&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Government capital endowment in region r</td>
<td>( R ) (7)</td>
</tr>
</tbody>
</table>

**TOTAL NUMBER OF VARIABLES:**
\[
18 \times R + 2 \times F \times R + 18 \times I \times R + 3 \times I \times R^2 + F \times I \times R + 2 \quad (1990)
\]

The figures in parentheses are the number of variables in a seven-region, six-sector model.

Table A.3

Definition of parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>te&lt;sub&gt;isz&lt;/sub&gt;</td>
<td>Sector export tax (subsidy) rate for goods to region r from region s</td>
</tr>
<tr>
<td>tm&lt;sub&gt;isz&lt;/sub&gt;</td>
<td>Sector tariff rate for goods from region s in region r</td>
</tr>
<tr>
<td>tind&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Sector indirect tax rate in region r</td>
</tr>
<tr>
<td>trc&lt;sub&gt;isz&lt;/sub&gt;</td>
<td>International transportation cost margin as percent value of f.o.b.</td>
</tr>
<tr>
<td>io&lt;sub&gt;ir&lt;/sub&gt;</td>
<td>Input/output coefficients for region r</td>
</tr>
<tr>
<td>kio&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Sector share of total investment in region r</td>
</tr>
<tr>
<td>pwt&lt;sub&gt;isz&lt;/sub&gt;</td>
<td>Sector weights of domestic goods supply at base year in region r</td>
</tr>
<tr>
<td>hdr&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Depreciation rate of private capital in region r</td>
</tr>
<tr>
<td>gdr&lt;sub&gt;r&lt;/sub&gt;</td>
<td>Depreciation rate of public capital in region r</td>
</tr>
</tbody>
</table>
Table A.3 (continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Gamma_{ir}$</td>
<td>Unit coefficients in first-level Arminton aggregation function</td>
</tr>
<tr>
<td>$\mu_{ir}$</td>
<td>Unit coefficients in second-level Arminton aggregation function of region $r$</td>
</tr>
<tr>
<td>$\alpha_{ir}$</td>
<td>Share parameters in the first-level Arminton aggregation function of region $r$</td>
</tr>
<tr>
<td>$\xi_{ir}$</td>
<td>Share parameters in the second-level Arminton aggregation function of region $r$</td>
</tr>
<tr>
<td>$\sigma_{mi}$</td>
<td>Substitution elasticities between domestic and import goods</td>
</tr>
<tr>
<td>$\sigma_{Li}$</td>
<td>Substitution elasticities among import goods from different regions</td>
</tr>
<tr>
<td>$\lambda_{ir}$</td>
<td>Unit coefficients in CET function of region $r$</td>
</tr>
<tr>
<td>$\kappa_{ir}$</td>
<td>Share parameters in CET function of region $r$</td>
</tr>
<tr>
<td>$\sigma_{ci}$</td>
<td>Elasticities of transformation between domestic sales and exports</td>
</tr>
<tr>
<td>$A_{ir}$</td>
<td>Unit parameter in value added function</td>
</tr>
<tr>
<td>$\delta_{ir}$</td>
<td>Factor share in value added function</td>
</tr>
<tr>
<td>$\sigma_{fr}$</td>
<td>Elasticities of substitution of factor in value added</td>
</tr>
<tr>
<td>$y_{ir}$</td>
<td>Sector minimum subsistence requirements for private household in region $r$</td>
</tr>
<tr>
<td>$\beta_{ir}$</td>
<td>Marginal propensity to consume for private household in region $r$</td>
</tr>
<tr>
<td>$\theta_{ir}$</td>
<td>Sector share of government spending in region $r$</td>
</tr>
<tr>
<td>$\tau_r$</td>
<td>Regional share of international shipping service supply</td>
</tr>
</tbody>
</table>

References


Codsi, G. and K.R. Pearson, GEMPACK: General-purpose software for applied general equilibrium and other economic modelers, Computer Science in Economics and Management 1, 189-207.


Krishnan, Ramayya, 1987, Knowledge based aids for model construction, Ph.D. dissertation
(University of Texas, Austin, TX).
Krishnan, Ramayya, David A. Kendrick, and Ronald M. Lee, 1988, A knowledge-based system for
production and distribution economics, Computer Science in Economics and Management 1,
53–72.
Lenard, Melanie L., 1993, An object-oriented approach to model management, Decision Support
Systems 9, 67–73.
Minsky, M.A., 1975, A framework for representing knowledge, in: P.H. Winston, ed., The psycho-
Pracht, William E., 1990, Model visualization: Graphical support for problem structuring and
knowledge organization, Decision Support System 6, 13–27.
Rumbaugh, James, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorensen,
Rutherford, Thomas F., 1989, General equilibrium modelling with MPS/GE, Mimeo. (Stanford
University, Stanford, CA).
Scarf, H.E., 1967, The approximation of fixed points of a continuous mapping, SIAM Journal of
Applied Mathematics 15, 1328–1343.
Shoven, J.B. and J. Whalley, 1973, General equilibrium with taxes: A computation procedure and
Shoven, J.B. and J. Whalley, 1974, On the computation of competitive equilibrium on international
Shoven, J.B. and J. Whalley, 1992, Applying general equilibrium, Cambridge surveys of economic
Slagle, J.R., 1990, One expert system shell, six expert systems, in: Proceedings of the fourth
international symposium on knowledge engineering (Barcelona, Spain) 16–29.