GOAL DIRECTED DISCOVERY AND EXPLANATION
IN PARTICLE PHYSICS

SAKIR KOCABAS*
Department of Artificial Intelligence
Tubitak - MRC, PK 21 Gebze, Turkey
Email: uckoca @ tritu.bitnet

Abstract: This paper describes a goal directed discovery system, TREV, which models the discovery of certain quantum properties and conservation laws by physicists in this century. The program is directed by consistency and completeness constraints, and has the capability of improving and revising its domain theory, and of explaining its knowledge state by these constraints. TREV is capable of formulating new elementary particles and particle reactions, and proposing observations to test their existence. The program can also generate exclusion hypotheses, and can revise its knowledge base in accordance with observational data.

1. Introduction

Computational modeling of discovery has been the focus of attention by several researchers in the last ten years, and a number of models with different capabilities have been developed. Among these systems, BACON (Langley, Simon, Bradshaw & Zytkowski, 1987), has the capabilities of data collection, quantitative reasoning and hypothesis formation; IDS (Nordhausen & Langley, 1993) and FAHRENHEIT (Zytkowski, 1987) have the features of data collection, qualitative and quantitative reasoning, and hypothesis formation; GLAUBER (Langley, et al., 1987), concept formation and the discovery of qualitative laws; STAHL (Zytkowski & Simon, 1986), STAHLp (Rose & Langley, 1986), REVOLVER (Rose & Langley, 1986), concept formation and theory revision; MECHAM (Valdes-Perez, 1992) discovery of reaction pathways; AbE (O'Rorke, Morris & Schulenburg, 1990), theory formation, explanation and theory revision; GALILEO (Zytkowski, 1990), theory formation; KEKADA (Kulkarni & Simon, 1988), goal selection, hypothesis formation, experiment design, and expectation setting; COAST (Rajamoney, 1990) and ECHO (Thagard, P. and Nowak, G., 1990), theory formation, theory revision and paradigm shifts by qualitative models; and BR-3 (Kocabas, 1991), theory formation and theory revision.

The subject of this paper is a goal directed discovery model TREV, with the capabilities of theory formation, experiment design, data acquisition, explanation, and theory revision. Before we describe the system and its behavior, it is appropriate to present some background information about its task domain, particle physics.

1.1. The Domain of Particle Physics

Until the last decade of the 19th century, material substances were thought to be consisting of indivisible atoms. Towards the end of that century, experiments with cathode ray tubes revealed the first elementary particle (the electron), which was to be identified as one of the basic components of an atom. Early in the 20th century, other elementary particles, the proton and the neutron were discovered. Later, observations on cosmic rays revealed a number of other particles such as the muon, pion, kaon, the neutrinos and the lambda particles. There are now well over a hundred elementary particles known, some of which are listed with their quantum properties in Table 1. Most of these particles are unstable, and quickly decay into a series of lighter and more stable particles such as the electron and neutrino, and into gamma rays. For example, a neutron decays to produce a proton, an elec-

* Also at: ITU, Faculty of Space Sciences and Technology, Istanbul, Turkey
tron and an antineutrino; and a pion decays into an antimuon and a neutrino:

\[ n \rightarrow p + e + \bar{\nu} \]
\[ \pi \rightarrow \mu + \nu. \]

Particles also interact with one another under natural and experimental conditions, producing other elementary particles or gamma radiation. These reactions are called "particle transmutations". An example to such interactions is the high-energy electron-proton collision, which produces a neutron and a neutrino:

\[ e + p \rightarrow n + \nu. \]

The theoretical possibility of such particle reactions depend on a series of quantum conservation laws. According to these laws, quantum properties such as electrical charge, spin, lepton number, baryon number, strangeness, energy, and momentum are conserved in particle decays and collisions. However, some quantum properties may not be conserved in certain reactions, (e.g., the strangeness property is not conserved in weak interactions.)

### 1.2. Theory Development in Particle Physics

The earliest known laws about elementary particle reactions were the energy and charge conservation laws. The law of the conservation of charge can be stated as follows: The sum of the charges of the initial particles entering a reaction is equal to the sum of the charges of the final particles. The following reactions conserve electrical charge and have been detected by physicists:

\[ p + p \rightarrow p + n + \pi \]
\[ \pi_0 \rightarrow \gamma + \gamma \]

where \( p, n, \pi, \pi_0, \) and gamma designate the proton, neutron, pion, pion-zero and gamma particles respectively. It has been known since early this century that the proton and electron have opposite and unit electrical charges. The neutron has been known to be unstable, decaying into a proton, an electron, and an antineutrino in what is called "beta decay", or

\[ n \rightarrow p + e + \bar{\nu} \]

but a proton decay has never been observed, and the stability of this particle had puzzled the physicists. Reactions such as

\[ p \rightarrow \pi + \pi_0 \]
\[ p \rightarrow e + \gamma \]

never happen despite the fact that they apparently obey the charge conservation law. A theoretical framework based only on the charge conservation law could not explain the absence of these reactions. In other words, such a theory would be incomplete concerning particle reactions.

Physicists resolved such problems by postulating new quantum properties and conservation laws, so that theoretically valid but physically unobservable reactions were rendered theoretically invalid by these laws (see, Omnes, 1970; Griffiths, 1987). In this way the absence of these reactions were explained by their violation of the conservation of the new quantum property. The next problem was to find the quantum value distribution of the new property over the elementary particles.

To illustrate how such conflicts were resolved, let us consider a reaction which conserves electrical charge but has not been observed.
Let us assume that this reaction violates the conservation of a new property (e.g., the "protonic charge"). Now, if we arbitrarily assign the new charge value to the proton as one and assume that the other particles, \( \pi \) and \( \pi_0 \), do not have this charge (i.e., they both have zero protonic charge), then the reaction would be unbalanced by the new charge (i.e., \( 1 =/= 0 + 0 \)). This would explain why the reaction had never been observed. Nevertheless, the value set \([1,0,0]\) is not the only one that makes the reaction unbalanced, as the values \([0,1,1]\), \([0,1,0]\), \([0,0,1]\) and \([1,1,1]\) produce the same effect. On the other hand, the new quantum values make some observed reactions unbalanced, as in the following reactions:

\[
\begin{align*}
 p + p & \rightarrow p + n + \pi \\
 p + \pi & \rightarrow n + \pi_0
\end{align*}
\]

These reactions conserve electrical charge, but not the "known" values of the new charge. This can be seen by substituting the protonic charge values:

\[
\begin{align*}
 1 + 1 &= 1 + n + 0 \\
 1 + \pi &= n + 0
\end{align*}
\]

This suggests that some of the other particles in these reactions must have nonzero protonic charge. Here, if we assign the protonic charge value of one to the neutron and zero to \( \pi \), the reactions would be balanced. However, other valid and observed reactions may conflict with the assigned values, and we may have to revise some of the assumptions about the protonic charge values of particles accordingly.

TREV, like its predecessor BR-3 (Kocabas, 1991) rediscovers the quantum properties in the same way as explained above. As the program's goal is to achieve a consistent and complete knowledge state, it postulates new hypotheses, and revises its domain knowledge until it achieves its goal state. In this way TREV models the discoveries of the lepton, baryon, electron, and muon number properties in particle physics. Apart from its theory formation and theory revision capabilities, the program also has the ability of proposing experiments and providing explanations for its assumptions about its domain objects.

In the remaining part of this paper we first present an overview of the system, and describe its behavior in modeling the discoveries of the quantum properties, in proposing experiments, and in providing explanations. This is followed by a discussion on the system's research goals, knowledge representation, theory revision and search methods, and its generality. The paper concludes with a summary of the results.

## 2. The System's Knowledge Representation and Behavior

The program uses a structured knowledge representation similar to qualitative schemas as in AbE (O’Rorke et al, 1990) and the other recent discovery models. This structured representation facilitates the system's identification of problem states such as incompleteness and inconsistency. Therefore we begin with describing the knowledge representation methods of TREV in some detail.

### 2.1. Knowledge Representation

TREV’s knowledge organization distinguishes descriptive and prescriptive knowledge. The former type of knowledge is represented as frames, and the latter as a series of operators and functions. The program has nine operators which are named as follows: evaluate, check-consistency, check-completeness, postulate-properties, revise-hypotheses, find-quantum-values, formulate-new-particles, formulate-virtual-particles, and formulate-reactions. The program also has a similarity based learning (SBL) module.

The main data items of TREV are elementary particles and their reactions. Both are represented as frames in the system's knowledge base. Particle frames include the name of the particle, the quantum properties and their values. The general form of a particle frame is as follows:

```
frame: P (frame name)
class = particle
q1 = v1
q2 = v2
...
qn = vn.
```

where P is the name of the particle, q1,...,qn the quantum properties, and v1,...,vn the corresponding quantum values, which can be -1, 0, or 1.
Particle reactions are represented in a similar way, this time containing information about the reactions, such as the particles involved, the reaction conditions, the physical status of the reaction, and its validity under the current theory. The general form of a particle reaction frame is as follows:

frame: reaction
class = physical event
actual status = A
logical status = L, logical-status(N,L)
reactants = R
products = P
active properties = Q, active-properties(N,Q)
reactants properties = Rp, reactants-properties(Q,Rp)
products properties = Pp, products-properties(Q,Pp)
conditions = (Rp = Pp) or (Rp ;~ Pp)

where A indicates whether the reaction has been physically observed or unobserved, and L indicates whether the reaction is valid or invalid under the current theoretical knowledge of the system. R and P are the lists of the particles involved in the reaction as the reactants and the products respectively. Q indicates the vector of quantum properties that play an active role in the reaction, while Rp and Pp are the quantum value vectors of the reactants and the products. Normally, particle reactions are added to the program’s knowledge base (e.g. for the reaction n → p + e + ν) as follows:

frame: rl
class = reaction
actual status = observed
reactants = [n ]
products = [p, e, ν ].

Such input reaction frames are then transformed into the form below by the 'evaluate' operator acting on the parent frame:

frame: rl,
class = reaction
actual status = observed
logical status = valid
reactants = [n ]
products = [p, e, ν ]
active properties = [q0, q1]
reactants properties = [1, 0]
products properties = [1, 0]
conditions = {[1,0] = [1,0]}. 

The amended slots are added after their values are calculated by the 'evaluate' operator.

2.2. Theory Formation and Revision

TREV has two operators for consistency and completeness checks: check-consistency and check-completeness. These operators can identify the problem states (inconsistency and incompleteness) about reactions. The check-consistency operator can decide whether the information in a reaction frame can be consistent or inconsistent with the system’s knowledge, by the following rules:

If R is a reaction,
and its actual status is observed,
and its logical status is valid,
then the system’s knowledge of R is consistent.

If R is a reaction,
and its actual status is observed,
and its logical status is invalid,
then the system’s knowledge of R is inconsistent.

The check-completeness operator, on the other hand, can also decide whether a reaction is explainable within the system’s current knowledge, i.e., why the reaction is physically observable or unobservable. In other words, the program can decide whether its knowledge concerning a particle reaction is complete or incomplete. The completeness rules are as follows:

If R is a reaction,
and its actual status is unobserved,
and its logical status is valid,
then the system’s knowledge of R is complete.

If R is a reaction,
and its actual status is unobserved,
and its logical status is invalid,
then the system’s knowledge of R is incomplete.

The program checks its knowledge about reactions for consistency and completeness every time it is presented with a new set of data, and tries to achieve a consistent and complete knowledge state. In this, TREV uses a control structure employed by its predecessor, BR-3 (Kocabas, 1991). Figure 1 summarizes the system’s con-
control structure. Accordingly, TREV first checks for consistency by using the above rules over its reaction frames, and reports inconsistent reactions to a message list.

An inconsistency report in the message list activates the revise-hypotheses operator. This operator modifies the system's knowledge about the particles' quantum property values by first turning the inconsistent reactions into algebraic equations, and then by finding sets of alternative quantum values for the particles appearing in these reactions. Since there are only three possible quantum values, namely -1, 0 and 1, modifications alternate between these values. Each value set is tried until the consistency constraints are satisfied.

On the other hand, after consistency has been achieved, if TREV cannot explain why a certain unobserved particle reaction is impossible, the program posts an incompleteness message to the message list. This in turn, activates the postulate-property operator, which postulates a new quantum property. The program adds the new quantum property to a new slot in the particle frames with the default values of zero.

The find-quantum-values operator turns the unobserved reaction formula into an algebraic inequality, and finds a set of quantum values for the particles in the formula. E.g. for the unobserved reaction \( p \rightarrow e + \gamma \), the inequalities

\[
0 \neq 0 + 1 \\
0 \neq -1 + 0 \\
1 \neq 0 + 0 \\
1 \neq -1 + 0 \\
1 \neq -1 + 1
\]

are generated by the program. Each of these inequalities represent a set of quantum values for the new property, which enable TREV to explain the absence of the reaction. The first quantum value set \((p = 0, e = 0, \gamma = 1)\) is assigned to the particles first. However, the new values must be consistent with the system's knowledge of elementary particles and their observed reactions. To secure this, the quantum values for the new property are assigned to other particles, such that its conservation is satisfied in the observed reactions. The check-consistency operator checks if the new values are consistent, and the revise-hypotheses operator revises them as necessary. This cycle continues until the system achieves a consistent and complete knowledge state.

2.3. Formulation of New Particles

The program can define new particles by making modifications on the values of quantum property slots of existing particle frames. For example, from the neutron's frame

<table>
<thead>
<tr>
<th>frame: n (neutron)</th>
</tr>
</thead>
<tbody>
<tr>
<td>class = particle</td>
</tr>
<tr>
<td>q1 = 0 (electrical charge)</td>
</tr>
<tr>
<td>q2 = 0 (lepton number)</td>
</tr>
<tr>
<td>q3 = 1 (baryon number)</td>
</tr>
</tbody>
</table>

a new particle can be defined by changing the q1 value to -1 to obtain the particle

<table>
<thead>
<tr>
<th>frame: pp1 (proposed particle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>class = proposed particle</td>
</tr>
<tr>
<td>q1 = -1 (electrical charge)</td>
</tr>
<tr>
<td>q2 = 0 (lepton number)</td>
</tr>
<tr>
<td>q3 = 1 (baryon number)</td>
</tr>
</tbody>
</table>

which, incidentally corresponds to anti-proton. The program proposes to make observations to check whether such postulated particles exist in nature. The important point about this exercise is that certain quantum property combinations never exist (e.g. particles having nonzero baryon and lepton values at the same time.) In fact, this observation had led to the development of the quark theory in particle physics in the 1960s.
After observations, if the proposed particle has been decided not to exist in nature then it is recorded as nonexistent particle e.g. as

frame: np1
  class = nonexistent particle
  q1 = v1
  q2 = v2
  q3 = v3

From its accumulated knowledge about existing elementary particles, TREV can construct hypotheses about the nonexistence of certain quantum value combinations, by an inductive method called exclusion based learning (Kocabas, 1989). These hypotheses state that particles with certain quantum property value combinations cannot exist. TREV can modify its exclusion hypotheses in view of the new knowledge about elementary particles. As soon as a new particle frame is created, the program checks its exclusion hypotheses to decide if the quantum values of the particle contradicts a hypothesis. If it does, the individual hypothesis is removed. The exclusion hypotheses are added to the system's knowledge base as frames:

frame: ep1,
  class = excluded q-composition
  q1 = v1
  q2 = v2
  q3 = #

which means that the quantum values v1 and v2 for the properties q1 and q2 respectively, cannot be possessed by an elementary particle.

2.4. Formulation of Virtual Particles and New Reactions

The program formulates particle decays and collisions by first defining a set of "virtual" particles. These are formulated simply by adding the vectors of quantum property values of two or three particles. An example to such virtual particles is the one that is formulated by adding the quantum values of the proton [1,0,1] and electron [-1,1,0], resulting in a proton-electron virtual particle with the quantum values of [0,1,1].

proton electron (proton-electron)
[1,0,1] + [-1,1,0] = [0,1,1]

In this way, a virtual particle with zero electrical charge, and with lepton and baryon numbers of 1 is defined. Such virtual particles are used in constructing particle decay and collision reactions. One such possible construction can be a neutron decay:

\[ n \rightarrow p + e \]

which, incidentally, is not a valid reaction, because it does not conserve the quantum values of lepton property, as quantum value vectors of the reactants and products are not equal, i.e., \([0,1,0] \neq [0,1,1]\). On the other hand, the reaction, which is obtained by using the neutron and the virtual particle proton-electron-antineutrino (p,e,\(\bar{\nu}\)),

\[ n \rightarrow p + e + \bar{\nu} \]

is a valid and observed reaction as it conserves all the three quantum properties, electrical charge, lepton and baryon numbers with the quantum value vectors of both sides being equal, i.e \([0,1,0] = [0,1,1]\).

Testing the reactions proposed by TREV may lead to the discovery of new quantum properties. If a proposed reaction is valid by the program's knowledge of quantum values, but cannot be observed, then this creates an incompleteness problem for the program. As has been described above, in such cases TREV postulates a new quantum property and tries to find a consistent and complete set of values for particles regarding the new property.

2.5. TREV's Methods of Explanation

The program uses its structured knowledge representation for producing explanations about the objects and events of its domain. Explanations are provided when the system is in a consistent and complete knowledge state.

TREV can explain why a certain proposed particle reaction is consistent or inconsistent with the system's knowledge about particle physics. In this type of explanations, the program uses the definition of consistency over the reaction in question.

The consistency (or validity) of a certain proposed reaction is explained by proving that the reaction conserves the quantum values that the program knows. If the reaction does not conserve these quantum values, then it is not inconsistent (or invalid). Consistency (or validity) of a reaction can easily be decided by checking its logical status slot, or by calculating and comparing the qua-
tum value vectors of the reactant and the resultant particles. For example, the reaction \( n \rightarrow p + e + \nu \) is consistent because the actual state slot of the reaction's frame says that the reaction has been observed, and the logical status slot says it is valid. If the reaction frame does not have such a slot, then the check-validity operator fires, which in turn finds if the reaction conserves the known quantum properties.

TREV can explain why a certain reaction is not observable by proving that it violates the conservation of a quantum property that it knows. Also, by using its completeness constraints, the program can explain why the impossibility of a certain unobserved reaction is or is not explainable within the program's domain theory. When the program cannot explain the absence of such a reaction by its domain theory, then it concludes that its knowledge about elementary particles is incomplete concerning the unobserved reaction. As has been described above, TREV resolves such problem states by postulating a new quantum property.

On the other hand, the program can also explain why there can be no particles with a certain set of quantum properties, by using its exclusion hypotheses for such explanations. For example, the exclusion hypothesis

\[
\text{frame: ep1,}
\]

\[
\begin{align*}
\text{class} & = \text{excluded q-composition} \\
q1 & = 1 \\
q2 & = 1 \\
q3 & = \#
\end{align*}
\]

explains why there cannot be a particle with the quantum values of \( q_1 = 1, q_2 = 1, \) and \( q_3 = 0. \)

The system's explanatory power increases as it discovers new quantum properties, and as the particle descriptions become more detailed by including new quantum property slots and values.

TREV can learn its consistency and completeness constraints by its similarity based learning (SBL) module. In learning a concept (e.g. consistent), the SBL module compares the positive instances of the concept (i.e. valid and observed reactions), and creates the definition of the concept. The system's consistency and completeness rules are created in this way.

3. Discussion on the System's Methods

TREV is a system that combines several features of a discovery model. Every discovery system, by definition, must have the ability to learn. The program has three distinct types of learning ability, namely inductive learning and learning by discovery. As described above, TREV learns its consistency and completeness constraints by similarity based learning, and its exclusion hypotheses, by exclusion based learning methods. The program also constructs its domain theory with its ability to learn by observation and by discovery. The former involves the formulation of new particles and reactions, and their subsequent comparison with the physical world. The latter takes place by postulating new quantum properties and assigning a set of corresponding quantum values to the particles.

An important feature of a discovery model is theory development, which itself can be divided in two tasks as theory formation and theory revision. TREV extends its domain theory by using its learning and discovery abilities, by adding exclusion hypotheses, by formulating its consistency and completeness constraints, and by postulating new quantum properties when faced with an incomplete knowledge state. When it is faced with an inconsistent knowledge state, the program revises its domain theory (i.e. knowledge about particles and their reactions) by using its consistency constraints together with general algebraic constraints.

In its theory development and theory revision activities based on the consistency and completeness constraints, the program works in an integrated way. However, the system's other task operators work independently and in an uncoordinated way. For example, the evaluate, formulate-new-particles, formulate-virtual-particles, and formulate-reactions operators are fired by an external agent (e.g. a user) independently. Similarly, explanation generating functions of the system are called on user demand and for specific purposes, such as in explaining why a particular particle or reaction is unobservable.

Also, the operators which formulate new particles and reactions are not constrained by domain dependent and general constrains. Hence, they operate in a relatively large search space. As a result, these operators can formulate uninteresting domain objects as well as the interesting ones.
TREV's explanation functions take advantage of the system's structural knowledge representation. The explanations provided are simple, and do not go deeper into the system's domain theory. However, the program can be improved in this direction.

The program's ability to formulate new objects means that it has the ability to propose observations to decide whether the formulated objects (i.e. elementary particles and reactions) exist in nature. Observation results are entered by the 'user'. There are a few discovery models, such as IDS (Nordhausen & langley, 1993) and FAHRENHEIT (Zytkow, 1987) that can directly receive data from their physical environment. In the domain of TREV however, experimental conditions are rather complex for any direct data acquisition.

The program has two types of theory revision capability. One is based on using the consistency constraints, and the other is theory revision by observational evidence. The former uses algebraic and domain constrints, and therefore, is less general, while the latter is based on observational results, and therefore simpler and more general.

Another shortcoming of the program is that the theory formation and revision operators are fired by a rule set whose conditions are determined by the message list. In other words, the control rules are hardwired, though an explanation based learning method could be used to learn such rules. We will address this problem in the future versions of the program.

4. Conclusions

One important problem in artificial intelligence is building models that integrate different methods of representation and learning. We have described a discovery system, directed by completeness and consistency constraints, with the capabilities of theory formation and theory revision, and with the ability of explaining its knowledge state by its domain constraints. The system is capable of formulating new elementary particles and particle reactions, and proposing observations to test their existence. The program has a certain degree of integration in its representation, learning and discovery methods, which can be further improved.

References


