Rediscovering Physics With BACON.3*

Pat Langley
Department of Psychology
Carnegie-Mellon University
Pittsburgh, Pennsylvania 152:3

BACON.3 is a production system that discovers empirical laws. The program uses a few simple heuristics to solve a broad range of tasks. These rules detect constancies and trends in data, and lead to the formulation of hypotheses and the definition of theoretical terms. BACON.3 represents data at varying levels of description, where the lowest have been directly observed and the highest correspond to hypotheses that explain everything so far observed. The system can also run and relate multiple experiments, collapse hypotheses with identical conditions, ignore differences between similar concepts, and discover and ignore irrelevant variables. BACON.3 has shown its generality by rediscovering versions of the Ideal gas law, Kepler's third law, Coulomb's law, Ohm's law, and Galileo's laws for the pendulum and constant acceleration.

1. INTRODUCTION

Centuries ago, early physicists such as Kepler and Galileo began to discover laws that described the physical world. In this paper I describe BACON.3, a program that rediscovers a number of these laws. I begin with an example of how one might discover the ideal gas law. Next, I consider the program's representation of laws and data. Following this, I examine the heuristics used by the system. I conclude with a discussion of BACON.3's generality.

MOLES	TEMP.	PRESSURE	VOLUME	PV
1	300	300000	0.008320	2496.0
1	300	400000	0.006240	2496.0
1	300	500000	0.004992	2496.0
1	310	300000	0.008597	2579.2
1	310	400000	0.006448	2579.2
1	310	500000	0.005158	2579.2
1	320	300000	0.008875	2662.4
1	320	400000	0.006656	2662.4
1	320	500000	0.005325	2662.4

TABLE 1. DATA OBEYING THE IDEAL GAS LAW.

2. AN EXAMPLE: THE IDEAL GAS LAW

The general law for ideal gases may be stated as $pV/nT = R$, where p is the pressure, T is the temperature in degrees Kelvin, n is the quantity of gas in moles, V is the volume, and R is the constant 8.32. Table 1 shows some of the data gathered by varying p and T when n is 1. Note that as the pressure increases, the volume decreases. Since a function of the form $pV = k$ would lead to such a trend, one might calculate the values of the product pV. In fact, the values of this term are different constants for different values of n and T, as shown in Table 2.

MOLES	TEMP.	PV	PV/T
1	300	2496.0	8.32
1	310	2579.2	8.32
1	320	2662.4	8.32
2	300	4992.0	16.64
2	310	5158.4	16.64
2	320	5324.8	16.64
3	300	7488.0	24.96
3	310	7737.6	24.96
3	320	7987.2	24.96

TABLE 2. SECOND LEVEL SUMMARY OF THE GAS LAW DATA.

Upon examining the first three rows of Table 2, one might note that the values of pV and the temperature increase together. Moreover, this relationship is linear with a slope of 8.32; since the intercept is 0, the slope term can be represented as pV/T. When the number of moles is varied, other linear relations are found with different slopes and the same intercept, as shown in Table 3.

MOLES	PV/T	PV/NT
1	8.32	8.32
2	16.64	8.32
3	24.96	8.32

TABLE 3. THIRD LEVEL SUMMARY OF THE IDEAL GAS DATA.

Finally, one might notice that the values of pV/T increase along with the number of moles. When the term pV/nT is defined and found to have the constant value 8.32, one has arrived at the ideal gas law.

*This research was supported in part by Grant HES75-22021 from the National Science Foundation, in part by NIMH Grant MH07722, and in part by ARPA Grant F44620-73-C0074. I would like to thank H. A. Simon, Eric Johnson, Marshall Atlas, Marilyn Mantei, Doug Lenat, and Robert Neches for discussions leading to the ideas in this paper.

3. LEVELS OF DESCRIPTION

As the reader may have guessed, BACON.3 rediscovers the ideal gas law in a manner much like the above. BACON.3 uses strategies very similar to those used by its precursor, BACON. 1 [1] yet BACON.3 can discover the gas law while its predecessor could not. BACON. 1 made a sharp distinction between the <u>data</u> it had observed and the <u>hypotheses</u> which explained those data. BACON.3 blurs the distinction between data and hypotheses by allowing various <u>levels of description.</u> In the new program, regularities in one level of description lead to the creation of a higher level of description.

Like the earlier program, BACON.3 is implemented as an OPS2 [2] production system. BACON.3 shares a number of heuristics with BACON. 1, though these have been generalized to deal with any level of description. Like BACON. 1, the new system defines theoretical terms like pV, pV/T, and pV/nT to describe its data parsimoniously. These heuristics and others are discussed in the following section.

4. THE HEURISTICS OF BACON3

The BACON.3 program consists of some 86 0PS2 productions. These can be divided into seven major sets, which I discuss below. The first four sets are held in common with the BACON. 1 system; the final three *are* additions required by the new representation and the tasks BACON.3 must handle.

The first set of productions is responsible for gathering directly observable data. Seven of these are responsible for gathering information from the user about the task to be considered. The remaining 10 productions gather data through a standard factorial design, varying first one independent term, then another.

The second set of 16 productions is responsible for noting regularities in the data collected by the first set. BAC0N.3's <u>constancy detectors</u> can deal with either symbolic or numerical data, and lead to the creation of higher level descriptions. The basic constancy detector is a simple restatement of the traditional inductive inference rule for making generalizations. Similar rules add conditions to newly created hypotheses. The program has primitive facilities for dealing with <u>near</u> constancies in noisy data; this is accomplished by redefining the LISP equal function to ignore small differences.

BACON.3'* <u>trend detectors</u> operate only on numerical data. Some of these notice increasing and decreasing monotonic trends between variables. These heuristics work in conjunction with other trend detectors that further analyze the data. One of these applies if the elope is constant, and leads to the definition of two new theoretical terms, the slope and the intercept of the line relating the two variables. Otherwise, a new term is defined as the product or *the* ratio of the variables, depending on the numbers involved.

After a theoretical term has been defined, a third set of 3 productions calculates the values of this term. Once calculated, these values are fair game for the regularity detectors. Defined terms are not distinguished from direct observables when noting regularities; it is this recursive ability to apply the same heuristics to concepts of increasing complexity which gives BACON.3 its power.

Before calculating the values of a new theoretical term, BACON.3 must make sure that the term is not equivalent to an existing concept. Accordingly, a fourth set of 22 productions decomposes the new term into its primitive components. If the definition of the new term is identical with an existing definition, the term is thrown out and other relations are considered.

Suppose BACON.3 has defined two intercept concepts for the ideal gas data. The values of the first, $\text{intercept}_{pv,t,1}$ are 0 when the number of moles is 1, while the values of the second, $\text{intercept}_{pv,t,2}$ 0 when the number of moles is 2. One would like BACON.3 to generalize at this point, but because the two intercepts are different terms, the constancy detector cannot be applied. BACON.3 notes such similar terms, and defines an <u>abstracted</u> term which ignores their differences. The values of the new term are copied from the originals, and the constancy detector is applied to the new data.

BACON.3 generates different descriptions to summarize different constancies. If two descriptions are found to have identical conditions, they are combined into a single structure; only 3 productions are devoted to this process. Once this has happened to a number of descriptions, the values of the dependent terms can be compared and regularities may emerge.

In rediscovering Galileo's law for pendulums, BACON.3 begins by varying the weight of the suspended object and the initial angle of the string. These variables are <u>irrelevant</u> to the period of the pendulum, but this is not obvious from the outset. BACON.3 draws on a final set of 8 productions for noting irrelevant terms. These modify the data gathering scheme so the values of the irrelevant terms are no longer varied.

Ideal Gas Law	$pV/nT = k_1$
Kepler's Third Law	$d^3(a - k_2 t)^2 = k_3$
Coulomb's Law	$Fd^2/q_1 q_2 = k_4$
Galileo's Laws	$dP^2/Lt^2 = k_5$
Ohm's Law	$Td^2/(lc - k_6 c) = k_7$

Table 4. Equations discovered by BACON.3

5. THE GENERALITY OF BACON.3

BACON.3 successfully rediscovered the five laws summarized in Table 4. These equations do not entirely do justice to BAC0N.3's discoveries. Along with omitting the conditions placed on some of the laws, only one equation is shown for each task, while a number were formulated for some. However, they do suggest the diversity of the laws the program generated from its data. The ability to define ratios and products leads to terms taken to a power, as in Coulomb's and Galileo's laws. The abstraction strategy allows the use of linear combinations in new terms, as In Ohm's law. Taken together, these two strategies lead to a version of Kepler's third law, in which the square of a linear combination plays a role.

Table 5 presents statistics on the relative complexity of the laws found by BACON.3. Three measures *are* used - the number of productions fired, the average size of working memory, and the average size of the conflict set. These measures are not completely correlated, but one trend is clear; the discovery of Ohm's law required much more computation than did the other tasks.

	IDEAL GAS	KEPLER	COULOMB	GALILEO	OHM
PRODS. FIRED	1203	1119	1297	1150	4593
WORKING MEMORY	55.9	111.0	119.7	66.6	241.9
CONFLICT SET	6.2	7.3	6.8	10.9	8.3

TABLE 5 RELATIVE COMPLEXITY OF THE FIVE TASKS

Closer analysis reveals some of the reasons for the complexity of this task. Table 6 presents some characteristics of the problem spaces for the five tasks. Since more terms were related in this law, more levels of description were needed to arrive at it. Moreover, two completely independent sets of laws were discovered In this run. These required a large number of theoretical terms, and a still greater number of terms which were considered and rejected.

	IDEAL GAS	KEPLER	COULOMB	GALILEO	OHM
SIZE OF DESCRIPT.	4	5	4	4.5	5
NUMBER OF DESCRIPT.	27	18	27	23	54
NUMBER OF LEVELS	4	3	4	3	5
DEFINED TERMS	8	17	16	9	31
REDUNDANT TERMS	2	1	12	7	84

TABLE 6. PROBLEM SPACE CHARACTERISTICS FOR THE TASKS.

How general were the heuristics of BACON.3? The heuristics borrowed from BACON. 1 were used in each of the tasks. Table 7 shows those tasks in which each of BACON.3'5 new heuristics were used. Most of the heuristics were used in multiple tasks, suggesting considerable generality for the rules. The single exception is misleading, since irrelevant variables could be added to each of the tasks.

	IDEAL GAS	KEPLER	COULOMB	GALILEO	OHM	TOTAL
GNORING DIFFS.	X	X	X		X	4
RRELEVANT VARS.				X		1
:OLL. DESCRIPT.	X	X	X	X	X	5
)IFF. EXPERS.		X		X		2

TABLE 7 USE OF BACON 3'S HEURISTICS

A general discovery system should be sensitive to the order in which it observes its data, but robust enough to arrive at equivalent laws regardless of the order. In the Galilean run reported in Table 4, the irrelevant variables weight and angle were varied first and immediately found to have no influence. In a second run where these variables were varied last, identical laws were eventually reached but the computation required was greater.

Modifying the order of relevant variables also affected the behavior of the system. In a second run on the ideal gas law, the number of moles was varied first, followed by the temperature, followed by the pressure. In the initial run, three major theoretical terms were generated - pV at level 1, pV/T at level 2, and pV/nT at level 3. In the second run, a different path was taken to the same conclusion - V/n was defined at the first level, V/nT at the second level, and pV/nT at the third level.

In summary, BACON.3 is a production system that can rediscover a number of laws from the history of physics. The system draws on a small number of strategies for finding regularities, defining terms, ignoring differences, collapsing hypotheses, and determining Irrelevant variables. Like its predecessor, BACON.3 is a general discovery system. One piece of evidence for this claim is that BACON.3 solved five different tasks using the same small set of heuristics. A second point in favor of BACON.3's generality was its ability to resolve tasks when the data were gathered in different orders. In conclusion, the progress to date has been encouraging, and suggests that more interesting discoveries lie ahead.

[1] Langley, P. BACON. 1: A general discovery system. In Proc. CSCSI, 1977, 173-180.

[2] Forgy, C. and McDermott, J. 0PS2 Manual. Pittsburgh, Pa.: Carnegie-Mellon University, Department of Computer Science, 1977.