Tort Liability for Artificial Intelligence and Expert Systems, 10 Computer L.J. 127 (1990)

George S. Cole

TORT LIABILITY FOR ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS

George S. Cole, Esq. *

TABLE OF CONTENTS

---

    * M.S. Computer Science 1987, Stanford University; J.D. 1981, University of Michigan; A.B. 1978, Stanford University. Mr. Cole has a law practice, with an emphasis in computer law, in Northern California.

I. PREFATORY REMARKS

Computers are now a significant facet of modern society. As their capabilities advance from basic number crunching into extraordinarily sophisticated symbolic manipulation, an area heretofore the proprietary domain of human beings, we must adapt to the social, legal and ethical consequences. The changes in computer applications seen to date are minuscule compared to those soon to be realized.

The past ten years saw amazing evolution and advancement in the

computer science area known as artificial intelligence, as well as the creation of its pragmatic brother, expert systems. Both areas are moving out of research laboratories and into the commercial world. Sooner or later, some individual will bring suit alleging that his injuries were caused by a step in the march of progress and will seek to go beyond remedies (or barriers to remedies) available under contracts or sales law. When the injury arises, not from the behavior of the individuals who created a contract or persuaded a customer to make a purchase, but from the behavior of those computer applications programs known as "artificial intelligence" or "expert systems," how will or should current theories of liability apply? This Article addresses these concerns.[1]

II. AN INTRODUCTION TO ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS (FOR LAWYERS AND LAYMEN) AND SOME PRELIMINARY LEGAL CONSIDERATIONS

A. PRELIMINARY DEFINITIONS

No single, universally-accepted, definition for artificial intelligence or expert system exists.[2] The author proposes the following definitions

---

1. For an overview of possible theories, see generally Lucash, *Legal Liability for Malfunction and Misuse of Expert Systems*, 18 SPECIAL INTEREST GROUP COMPUTER-HUMAN INTERACTION BULL. 35 (1986); Wilson, *Who Should Pay for Computer Error?*, COMPUTERWORLD, Sept. 17, 1984, at 20. Theories which focus on the potentially tortious behavior of the humans buying or selling the applications, such as breach of contract, warranties, fraud, or negligent misrepresentation, are not covered in this Article.

2. The latter may be easier to define, if only because we have an imperfect understanding of natural intelligence. Dr. Austin Henderson at Xerox PARC, in a relaxed discussion, agreed with this author that a sophisticated spreadsheet arguably qualifies as an artificial intelligence, inasmuch as it replicates human intelligence in formula transactions interacting with the input numbers, or as an expert system in the domain of numerical expressions and manipulation of mathematical formulae. However, the lack of ability to modify the world, or of internalized knowledge about the meaning of the formulae and the outputs' designated interpretations, disqualifies the spreadsheet as either AI or ES. Dr. Hubert Dreyfus argues that no true artificial intelligence exists or ever will exist. Dreyfus, *Why Computers Can't Be Intelligent*, CREATIVE COMPUTING, Mar. 1980, at 72-78.

See also 1 THE HANDBOOK OF ARTIFICIAL INTELLIGENCE, ch. 1, at 3 (vol. 1 & vol. 2: A. Barr & E. Feigenbaum eds.; vol. 3: P. Cohen & E. Feigenbaum eds. 1981) [hereinafter HANDBOOK]; P. WINSTON, ARTIFICIAL INTELLIGENCE (2d ed. 1984). *Cf.* M. WEIK, STANDARD DICTIONARY OF COMPUTERS AND INFORMATION PROCESSING 192 (rev. 2d ed. 1977).

The Defense Advanced Projects Research Agency defined an "expert system" as "the codification of any process that people use to reason, plan, or make decisions as a set of computer rules . . . [involving] a detailed description of the precise thought processes used." DEFENSE ADVANCED PROJECTS RESEARCH AGENCY (DAPRA), STRATEGIC COMPUTING 7 (Nat'l Technology Information Serv., Rep. No. AD-A141 982/9; 19-449, 503 PC A06/MF A01, Oct. 1983).

Dr. Bruce Buchanan, in his review of the expert system in the field and in search literature, proposed a four-dimensional evaluation scheme whereby each expert system

for evaluating the applicable theories of liability considered in this Article. Artificial Intelligence (AI) describes a member of the class of programs that emulate knowledgeable manipulation of the real world. AI involves sensing, reasoning, and interaction within the real world, or at least a portion thereof; the effect of the program's outputs may become part of the program's later inputs.[3] AI implements many research problems including, but not limited to, machine vision (or other sensing), robotics, and learning.

Expert system (ES) describes a member of the class of programs that emulate (or even excel) human experts' performance in a prior specified domain of knowledge. An ES reasons by manipulating abstract symbology rather than the external or real world. The human user must interpret the output, and the output does not interact directly with later inputs to the program. While both AI and ES programs use techniques drawn from current state of the art research in AI, for knowledge representation and implementation of the transformation and control processes employed by the program, each ES is designed only to act sensibly within its program-specific domain.[4] The crucial hallmark of AI is any interactive feedback with the real world which connects its behavior and effect in the real world to its inputs; the crucial hallmark of ES is any internalized model of the effects of successfully concluding its computations, i.e., any domain knowledge.[5]

---

would be measured by its AI methodology, high performance, flexibility, and understandability. *See* B. BUCHANAN, EXPERT SYSTEMS: WORKING SYSTEMS AND THE RESEARCH LITERATURE (Knowledge Sys. Lab., Dept. Computer Sci., Stanford Univ., Report No. KSL-85-37, 1985).

There are two primary source books for practitioners and researchers. BUILDING EXPERT SYSTEMS (F. Hayes-Roth, D. Waterman & D. Lenat eds. 1983) [hereinafter BUILDING ES] (articles were written by a substantial portion (50) of the recognized practitioners in that field); READINGS IN KNOWLEDGE REPRESENTATION (R. Brachman & H. Levesque eds. 1985) [hereinafter READINGS IN KR]. The former text states that expert or knowledge-based systems "are designed to represent and apply factual knowledge of specific areas of expertise to solve problems." BUILDING ES, *supra*, at xi.

3. Because the real world is involved in this feedback loop, there is always the possibility of unpredicted transformations or interactions. Thus, there is an inherent, inescapable possibility of failure built in.

4. For example, a financial planning ES manipulates numbers with a default assumption that the numbers represent dollar values. Negative, fractional, irrational, or imaginary numbers become far less sensible within that domain. A spreadsheet, which may be used for engineering, architectural, or production and inventory calculations or for physics or mathematics problem-solving, is far more general-purpose and does not have the implicit domain stated in its description.

5. For example, assume three automobile repair programs exist. One simply produces readouts of the exhaust emissions. A second, given readings (indirectly through a mechanic typing on a keyboard, or directly from metering devices), produces a reasoned diagnosis. A third goes further and adjusts the timing or reprograms the automobile's on-board computer. In an attempt to meet statutory requirements, the first would be classi-

The behavior of these complex programs must not only be evaluated according to the circumstances of the injury, but also according to the environment in which the programs were operating when the injury was caused, and the circumstances and processes surrounding the programs' creation. The next three sections of this Article examine the basic environment and characteristics of modern AI and ES programs. The fourth and fifth sections briefly discuss some legal assumptions and concerns; they also indicate the issues to be covered by the remainder of the Article. Lawyers may find the next three sections particularly helpful, especially as a source for further references; computer scientists may find the subsequent two portions helpful as a guide to the manifold concerns and approaches of lawyers.

B. THE ENVIRONMENT AND REASONING METHODS
UNDERLYING AI AND ES

1. *The Environment of AI and ES*

AI and ES exist atop sophisticated and complex computer environments composed of supporting hardware and software. All computer operations depend upon the sequencing and state of the electronic impulses within the hardware—a compilation of a central processor, a memory unit, and input/output devices. These electronic impulses can be distorted by background radiation, metal fatigue, or other physical interruptions, causing unpredicted alterations, i.e., errors.[6] The denser the computer (in memory size or processing power), or the harder the program (in processing time and memory requirements), the more physical errors are likely to occur. While redundancy and self-checking capacity can be built in to these machines, they are still subject to "Murphy's Law,"[7] just as the humans who build them. AI and ES generally require larger machines, more memory, and more complex input/output devices because of their inherent complexity and large database requirements. Thus, they are more likely to encounter such errors.

The electronic impulses in the hardware are manipulated by

---

fied as a standard program, the second as an expert system, and the third as artificial intelligence.

6. In a production-operating system, 11% of all software errors and 40% of all software failures were computer-hardware related. *See also* N. LEVENSON, SOFTWARE SAFETY: WHY, WHAT AND HOW 7 (1986) (Dept. Info. & Computer Sci. at Univ. Cal. Irvine Rep. No. TR-86-04, 1986) [hereinafter LEVENSON] (citing Iyer & Velardi, *Hardware Related Software Errors: Measurement and Analysis*, TRANSACTIONS ON SOFTWARE ENG'G, Feb. 1985, at 223).

7. Adding redundancy and error-checking routines further complicates the program, allowing more room for error, as evidenced by the synchronization failures among the on-board computers in the first space shuttle flight. LEVENSON, *supra* note 6, at 12 (citing Garman, *The Bug Heard "Round the World"*, 6 ACM SOFTWARE ENG'G NOTES 3 (1981)).

software in a hierarchy of increasing abstraction and decreasing detail. At the bottom of the hierarchy is the operating system which directs the electronic impulses according to the hardware's connectivity. It arbitrates processing, memory access, and input/output demands. At this level, multiple classes of conditions which are not reproducible, and which are possibly unique, can rise to errors.[8]

Above the operating system is the assembler, which translates the contents of the memory registers (which can be either data or instructions) into the sequence of machine code which is processed by the operating system.[9] Compilers translate application programs into assembly language and are designed to permit some machines to use a particular high-level language without a great efficiency loss. These application programs are usually written with some idea of the limitations (or strengths) of the particular hardware and software beneath them, and they attempt to capture the expressive range of the next level above.[10] An alternative, at this stage, is an interpreter, which directly translates a statement in a high-level language to machine readable form directly; interpreted code is usually less efficient than compiled code, but it allows for direct feedback and rapid implementation.

Most application programs are written in a high-level language, e.g., Ada, Pascal, LISP, C, or their more primitive predecessors, BASIC and FORTRAN.[11] An application program is presumed to be somewhat implementation-independent; high quality programming includes docu-

---

8. For example: (1) The internal state (such as the number of users, register values, time or date information) was not determined initially; (2) There are undetermined language semantics for some operations, such as the outcome of a race condition (two processes accessing or providing the same register or memory location); (3) Hardware concerns—heat transfer, cable length, pin locations—can produce unforeseen software faults; and (4) Timing errors can arise that are invisible and untraceable due to the Heisenberg effect; the process of tracing steals enough instruction cycles to nullify the problem. Operating systems traditionally provide and suffer from a large number of such bugs. Interviews with Michael Dixon, Consultant, Xerox PARC, and Frank Halasz, Senior Researcher, PARC (now with MCC).

9. This unanimity of program and data comprise the theoretical "Von Neumann machine," and distinguish assembler code from operating system code—the latter must know which string of impulses is data and which is instruction. An often criticized programming "hack" is to write self-modifying code, i.e., an assembly program that treats its instructions as data, changes them, and runs the new instructions. This interplay between data and instructions lies behind core wars, viruses, Trojan Horses, and other potentially devastating programming techniques.

10. Compilers were the first automatic programming development in the 1950s. *See also* 2 HANDBOOK, *supra* note 2, ch. 10, at 297. *See generally* A. AHO, R. SETHI & J. ULLMAN, COMPILERS: PRINCIPLES, TECHNIQUES, TOOLS (1986) [hereinafter COMPILERS].

11. There are a host of languages and dialects. In data processing markets, COBOL and later SNOBOL developed. In addition, MacLisp, Interlisp-D, FranzLisp, UCSDLisp, GoldonCommonLisp, and CommonLisp all exist today. One of the greatest problems for the industry is attaining a national or international standard for any high level language

mentation of any deviation from an existing standard that was devised
to take advantage of a particular compiling, assembling, or hardware en-
vironment. The specific techniques or algorithms used in any applica-
tion may depend upon the high-level language's syntax and semantics.
Although, in theory, most programs could be implemented in any lan-
guage, in practice, different languages have different strengths and
weaknesses. The more complex an application program becomes, or the
more it requires interaction among multiple authors or a long period of
time to develop, the more likely a specification language exists which
states inputs, outputs, desired domain behavior, and (possibly) proce-
dural behavior. The specification language may influence the design
chosen for implementation in the high-level language.[12] At the top of
the hierarchy is the "interaction language" which allows the program to
(1) interact, at run-time, with the world, (2) read input, and (3) write
output.[13]

2. *Environmental Problem Sources*

Every program below AI or ES in the program hierarchy can have
undetected programming errors[14] which may interact in unforeseen
ways.[15] Syntax and semantic errors are the most readily eliminated be-
cause they may be tested against the constraints that are generalized

---

which will specify all compilers, assemblers, operating systems, and hardware necessary
to support the high-level language.

12. AI research into automatic programming is examining these specification lan-
guages as one step (possibly out of many) between current programming languages and
natural language. 2 HANDBOOK, *supra* note 2, ch. 10, at 297.

13. These words should be understood as expressing generalized interaction with the
world, i.e., "reading" includes getting a temperature from a thermostat and "writing" in-
cludes turning on a furnace. The range of possible terms, functions, and expressions re-
quiring comprehension is thus domain-specific. The epistemological debate over how
much any computer understands its world begins with a discussion of "reading" and
"writing" and moves further into theory. The practical reality is AI and ES programs do
interact within a domain-specific range in the real world.

14. Typical programming classes separate errors into multiple classes: (1) syntax er-
rors (use of impossible character combinations, e.g., A =: B in Pascal); (2) semantic errors
(using reserved or control words incorrectly or using the wrong control word, such as
"do" rather than "for"); (3) intent errors (using an "until" rather than a "while," i.e., test-
ing after the process rather than before or during each interaction); (4) design errors
(leaving out some step); and (5) specification errors (solving the wrong problem).

15. If, for example, the application were designed to allow unlimited size of variable
names (presuming that the compiler knew of indirect referencing methods to cope with
machine limitations), but the purchaser's compiler contained (as an optimization, perhaps)
a limited size of variable names, unexpected errors could arise. One rule dealing with Na-
poleon Bonaparte and a second rule dealing with Napoleon Desserts may become con-
fused. *See* COMPILERS, *supra* note 10, at 429-42; M. CLANCY & D. COOPER, OH! PASCAL! 25
(2d ed. 1985) [hereinafter OH! PASCAL!]; Gemignani, *Product Liability and Software*, 8
RUTGERS COMPUTER & TECH. L.J. 173, 200 n.88 (1981).

across the entire language and working conditions.[16] While context-independent errors are potentially eliminable, one must take care to discriminate between true syntactic errors, and ones whose effects are semantic.[17]

Research into the techniques and limitations of automatic mathematical program verification is ongoing.[18] This offers hope that occurrences of the first two classes of program errors may eventually be diminished drastically. But it is impossible to hope that any progress can be made in the vast remaining classes.[19] At present, the state of the art is far behind the skill and techniques necessary to make automatic verification even potentially reliable.[20] The size of the programs in the

---

16. New tools include: (i) compilers with automatic or interactive syntax correction to catch typing mistakes or semantic checks (e.g., a Pascal program that makes an assignment across incompatible types will not compile); (ii) on-line assistance debuggers or traces (programs that allow the program to step through a program instruction by instruction until a break is encountered, and then provide a snapshot of the values of the variables and context of the program); and (iii) environmentally intelligent, context-sensitive programs that trap errors and attempt repairs. For example, with a LISP environment on a Dandelion machine which has the DWIM function ("Do What I Meant"), it attempts to interpret a not quite understood command. A typical translation is "logout" to "= LOGOUT," then it logs the user off the machine. It is not foolproof; when there are more serious problems, the user may have to guess how to respond to messages such as "Action unknown. Type Control-P to muddle through . . ." or whether to "scavenge" as suggested. *See infra* note 87 and accompanying text.

17. *See infra* note 315 and accompanying text. Teaching advances include a better understanding of the underlying science and of the importance of the human factors in both programming and use of computers, as well as indoctrination into techniques designed to help catch errors: modular programming, "stub" testing of modules, and fault-point analysis as well as goal-point analysis.

18. *See* 2 HANDBOOK, *supra* note 2, at ch. 10. Manna & Waldinger, *A Deductive Approach to Program Synthesis*, in READINGS IN ARTIFICIAL INTELLIGENCE 141 (B. Webber & N. Nilsson eds. 1982) [hereinafter READINGS IN AI].

19. *Cf.* LEVENSON, *supra* note 6, at 13, 23-32 (some alternative techniques). Consider the difficulties inherent in proving intent in a legal setting.

20. Software testing workshops are sponsored every four years by the Association for Computing Machinery/SIGSOFT and IEEE/CS Software Engineering Technical Committee. In the 1986 conference, eighteen papers were presented during six working sessions; the major theme was how thorough a test set of questions or text coverage for a program should be. (The number of errors detected cannot be defined as a percent of total errors as the set of errors is unknown.) Testing approaches included mutation testing (making random modifications to the program) and path analysis. Random testing was ruled out as the probability of hitting extremes (using uniform distribution) was negligible. The programs considered were occasionally three-hundred lines long (an improvement over prior experiments which considered seventeen-line programs)—a far cry from modern complex programs which are thousands or tens of thousands of lines long.

The chief problem recorded by one participant was that the typical model (i.e., the computation model of a flow chart of function computation, where procedures were infinite graphs with operations or tests at the nodes and fixed sets of program variables which contained disjoint values) was outdated. Modern techniques such as data structure

underlying software environment, let alone AI or ES, is too great for
this approach to currently be practicable.[21] It is a truism that program-
ming errors are inevitable:

> Specific programs can be tested to reveal 'bugs' that must be corrected,
> however, no amount of testing can guarantee that all the bugs in the
> program have been found; even after extensive testing, the program,
> which in a major software project can be extremely complicated and
> expensive, may still fail miserably.[22]

Within the software domain, a solution may turn out to be worse
than the problem. The complexity of interactions between levels can be
so great that even when an error in one program at one level is uncov-
ered (typically through a run-time error), any attempt to fix the bug
creates more errors to be discovered later.[23]

3. *Reasoning Methods Underlying AI and ES*

The basic process for all computer programs is transformation of
inputs into outputs.[24] The transition between internal electronic im-

manipulation involving allocated records and pointer manipulation (the recognition that a
significant part of the computation state might be represented by a call stack of proce-
dures, or even of recognition) were not in the model, nor was a collection of tiny proce-
dures calling each other incestuously such as those one might find in LISP programming
or in PROLOG-based AI or ES. Letter from Howard Sturgis (Xerox PARC) to the au-
thor (July 25, 1986).

21. The space shuttle ground processing system, with over a half million lines of code,
is one of the largest real-time systems ever developed. The stable release version under-
went 2,177 hours of simulation testing and then 280 hours of actual use during the third
shuttle mission. Yet, the mission uncovered one critical, three major, and twenty minor
errors. *See Development of Software for Ballistic-Missile Defense*, SCI. AM., Dec. 1985, at
47-48 (compares the difficulty of specifying the strategic defense initiative programming
requirements [twelve to twenty times the size of the shuttle's code] and, the difficulty of
specifying an implementation of the tax code: the former would occupy tens of thousands
of pages in the specification language; the latter, approximately 3,000 pages). *See id.* at 47-
53 (a good review of the basic problems that can arise in any software development
project).

22. Gemignani, *supra* note 15, at 269-313 (footnote omitted). *See infra* text accompa-
nying note 183 (discussing the inevitability of error). Although modern programming lan-
guages, techniques, training, and environments make many formerly common mistakes
less likely to appear, or catch the mistakes in the early stages, there seems little doubt
that the basic concept (if not the numbers) of the inevitability of erroneous programming
is still valid today.

23. The probability that attempts to remove program errors introduces one or more
additional errors varies; estimates range from fifteen to fifty percent. Experience with
large control programs—those having between 100,000 and 2,000,000 lines—suggests that
the chance of introducing a severe error during the correction of original errors is large
enough that only a small fraction of the original errors should be corrected. Adams, *Opti-
mizing Preventive Service of Software Products*, 28 IBM J. RES. & DEV. 8, 12 (1984). *See
also* Misra, *Software Reliability Analysis*, 22 IBM SYS. J. 262 (1983).

24. *See* J. HOPCROFT & J. ULLMAN, INTRODUCTION TO AUTOMATA THEORY, LAN-

pulses and the real world requires assigning an interpretation or meaning to all possible states of the computer and its input/output devices.[25] The entire hierarchy of software accomplishes specification of how these transitions take place.[26] Many transitions are possible: mathematical, logical, linguistic, symbolic, relational—but most can be expressed via interpretations of the first two. The methods and limitations of mathematical and logical transitions in computer science will be examined in turn.

a. Mathematical calculation.

All implementations of mathematical computations (addition, subtraction, multiplication, division, exponentiation, trigonometric functions, calculus, fast Fourier transforms, matrix algebra, and other, more abstract, functions) are based on the operations of counting and comparison. The programmer must establish a technique that translates the higher mathematical formulae into these processes.[27] The use of different types of hardware allow the use of different techniques, with lesser or greater efficiency.[28]

b. Logical reasoning.

Logic is a mathematics amenable to manipulation of many more concepts than numbers via symbolic representation and interpretation.[29] The impulses in the computer's hardware become the representation of "true" or "false" values for facts and knowledge, and the reasoning or inference operations become mathematical operations.[30]

GUAGES, AND COMPUTATION (1979) [hereinafter HOPCROFT]. Note that either or both input and output classes may include a null element.

25. *See* D. HOFSTADTER, GÖDEL, ESCHER, BACH: AN ETERNAL GOLDEN BRAID 49-52 (1979). Douglas R. Hofstadter gives a good example of how the interpretation of a symbol can create or deny meaning. He points out that "2 P 3 E 5" can mean "2 plus 3 equals 5," or "2 is produced when 3 is eliminated from 5". The license plate "6E" can mean "sexy," an apartment number, or "110" (in hexadecimal). When considering assignments between possible computer states and meanings, remember Murphy's law and realize that the largest equivalence class is probably that containing all the error states. *Id.*

26. The state of the individual bits is typically represented by a string of 1's and 0's; in an eight-bit (byte) register, "00000001" may mean the positive number one, "true," "yes," or "memory address 1." Relationships between truth, language, and the world were explored by Alfred Tarski. *See* A. TARSKI, LOGIC, SEMANTICS, METAMATHEMATICS, PAPERS FROM 1923 TO 1938 (J. Woodger trans. 1956).

27. A general source book is R. SEDGEWICK, ALGORITHMS (1983).

28. The "supercomputers," such as CRAY-I, allow direct counting of a larger number of bit locations, both along the width or power dimension and along the depth or sum dimension.

29. This is a pragmatic, rather than a theoretically pure statement. For a complete refutation, see HOFSTADTER, *supra* note 25.

30. Computer logic is typically Boolean and implemented by truth tables:

Deducted representations (in the form of "facts") within the logical process are assigned impulses as the process of inference—these mathematical operations—takes place. As with other mathematical operations, comparison and counting perform the transformations.

The most common logical reasoning process is based on predicate calculus or a subset thereof,[31] the focus of the fifth generation. The inference technique most used is resolution.[32] Programmers currently

P	Q	P—>Q
T	T	T
T	F	F
F	T	T
F	F	T

This is the truth table for the implications operator, or "if." The values of the two operands are compared and the result is assigned to a third. Thus, the first line of the table is read as follows: given that P is "true" and given that Q is "true," then the operation of "if P, then Q" produces the result of "true." Contrary to common sense, the implication of a false antecedent (P) is always true because anything can be proven from a false assumption. *See also* Allen, *Formalizing Hohfeldian Analysis to Clarify the Multiple Senses of "Legal Right": A Powerful Lens for the Electronic Age*, 48 S. CAL. L. REV. 428 (1974); Keuth, *On Some Logical Characteristics of Legal Norms*, 15 JURIMETRICS J. 160 (1977).

31. Predicate calculus is also known as the "Fregean" or "propositional" form of representation, so called by its inventor, Gottlob Frege. Atoms, individual elements, or place-holding variables (memory locations) are used (perhaps with transformational functions, i.e., subordinate processes) to build terms, which can be structured with logical operands into sentences. Assume Bill, Sam, and Mary are atoms; a, b, and c are constants; x, y, and z are variables; *FATHER-OF* and *SISTER-OF* are functions); MALE and FATHER are predicates; and &,v, and → (and, or, if-then a.k.a. implication) are logical operands. MALE (Bill) is true while MALE (Mary) is false and MALE (x) may be either true or false, depending on what atom x represents. From a = Sam, *SISTER-OF* (a) = Mary, and *FATHER-OF* (Mary) = Bill, we can reason that *FATHER-OF* (Sam) = Bill.

AI or ES could not reach our conclusion without a rule enabling that inference, which might be expressed as *SISTER-OF* (x) = by & *FATHER-OF* (y) (→ *FATHER-OF* (x) = bz). Such a rule, of course, does not account for step-children; further rules are needed. *See also* 1 HANDBOOK, *supra* note 2, at ch. 3, 13; Begriffsschrift, *A Formula Language Modelled Upon That of Arithmetic, for Pure Thought*, in FROM FREGE TO GÖDEL: A SOURCE BOOK IN MATHEMATICAL LANGUAGE, 1879-1931 1 (1967); 1 Z. MANNA & R. WALD-INGER, THE LOGICAL BASIS FOR COMPUTER PROGRAMMING (1985)[hereinafter LOGICAL BASIS]; Creary, *Propositional Attitudes: Fregean Representation and Simulative Reasoning*, 1979 PROC. SIXTH INT'L JOINT CONF. ARTIFICIAL INTELLIGENCE 1 (available at Computer Science Department, Stanford University).

The standard form of PROLOG allows only a form of predicate calculus known as "Horn clauses," whereby only the first operand or predicate may be positive (or "true"). *See* Warren, Pereira & Pereira, *PROLOG—The Language and Its Implementation Compared With LISP*, 1977 PROC. SYMP. ARTIFICIAL INTELLIGENCE & PROGRAMMING LANGUAGES (Association for Computing Machinery) 109.

32. Resolution negates the goal sought to be proved and attempts to derive a logical

favor resolution because, given certain constraints on the operations used to perform the transformations, i.e., on the inference techniques, it can be mathematically established to be both correct and complete.[33] "Correct" means that the process of resolving literals against the sentences will not introduce an inconsistency; "complete" means that if an answer can be found, it will be found. Research into which constraints are necessary, as opposed to sufficient, is ongoing, as is research into increasingly efficient implementations of all of the problems involved with mechanical representation and manipulation of the full power of predicate calculus.

4. Reasoning Methods' Problem Sources

a. Mathematical calculation.

Even simple addition can cause an error because each machine has an inherent limitation on the number of distinct states which it can count or perceive, based on the size of its registers whose contents reflect powers of 2 (two). This limit is independent of the task performed, but may affect the resulting outputs. Consequently, recognizing that an

---

inconsistency. For example, if the background knowledge is the following three sentences (the minus sign stands for "not"):

    (1) -LUCKY (x) v HAPPY (x) (anyone is not lucky or they are happy)
    (2) -HAPPY (x) v RICH (x) (anyone is not happy or they are rich)
    (3) LUCKY (Sam)
       (Sam is lucky).

If the goal is to prove RICH(Sam), resolution works as follows: First, assume (state) the negation of the goal: RICH(Sam) This statement is compared against (2). When x is instantiated with Sam (and the instantiation has to be identical for all occurrences of the variable in the same sentence), there is a contradiction (Sam cannot be both RICH and -RICH). This is resolved by dropping the inconsistent portion of that sentence and concluding: -HAPPY(Sam). This new sentence is compared against sentence (1). There is a new contradiction, which is resolved by dropping the inconsistent portion, and so the conclusion is -LUCKY(Sam). Finally, when this new statement is compared against sentence (3), there is a contradiction that resolves in an empty sentence, or NIL. The meaning of this empty sentence under resolution is that the original unnegated sentence is proved true, because its contradiction cannot be true. Resolution was introduced by Robinson in 1965. Robinson, *A Machine-Oriented Logic Based on the Resolution Principle*, 12 J. A. COMPUTING MACHINERY 23 (1965). *See* 3 HANDBOOK, *supra* note 2, ch. 12, at 86-94. *See also* N. NILSSON & M. GENESERETH, LOGICAL FOUNDATIONS OF ARTIFICIAL INTELLIGENCE 63 (1987) [hereinafter LOGICAL FOUNDATIONS].

    33. Resolution requires a simplified version of predicate calculus called "clausal form" in which the sentences are expressed as sets of literals and clauses comprised of disjunctions of literals. There are machine-usable rules for translation of predicate calculus into clausal form. *See* LOGICAL FOUNDATIONS, *supra* note 32, at 64. As the authors point out, resolution is refutation complete; it cannot generate every sentence implied by its starting data. For example, it does not generate all the possible tautologies, but it will discover whether or not the given data plus the goal contain an inconsistency. Furthermore, it is correct insofar as it will not introduce an inconsistency where it did not exist before.

answer is wrong requires both a knowledge of the domain, as well as a knowledge of the process whereby the answer was gained.[34]

Arithmetic implementations are not only subject to hardware limitations but to accuracy limits selected by the programmer as well.[35] Since hardware limitations are invisible to any program running on that machine (except as a source of run-time errors), the absence of a provision to check interim results may mean that the final results reflect the computer's (not the process') capacity for discrimination.[36]

b. *Logical reasoning.*

Typical implementations of logic do not require a true antecedent to pursue a line of reasoning, nor do they have the means to distinguish between hypothetical and real worlds.[37] Moreover, they do not all ensure (without more) that the limits of unification and differentiation between variables in bound or free logical sentences are kept,[38] nor that full equivalence of logical sentences is utilized.[39]

---

34. If the largest register in a computer's arithmetic-logic unit has 16 gates, typically it will recognize only integers between -32,768 and +32,767 (positive and negative 2^{15} with one slot for zero). Counting beyond either limit typically causes an error. If the user is fortunate, an error message such as "ERROR—Register Overflow" appears, which interrupts the program and, thus, does not allow it to continue with an improper result. *But see infra* note 295 and accompanying text, for another possible response and the legal implication there.

35. Mathematical function programs are often handed about, but the limitations may be improperly documented. *See infra* note 310, for an example of the type of problem that can arise from differing limits on accuracy.

36. More sophisticated programs (and computers) work around these limitations by transferring arithmetical computations that may exceed hardware limits to a sub-program that implements more complex and detailed mathematical formulae. Such mathematical calculations (BIGNUMS) use sums of powers of 2, including negative powers or fractions, and are finer than those typically produced by people; nonetheless, they may represent approximations.

37. With the recent advent of unfortunately named "automatic truth maintenance systems" (ATMS reasoning), the distinction between hypotheses or one hypothesis and the real world becomes possible. *See* Doyle, *Truth Maintenance Systems for Problem Solving*, MIT AI TECH. REP., Jan. 1978, at 419; deKleer, *An Assumption-Based TMS*, 28 ARTIFICIAL INTELLIGENCE 127 (1986); deKleer, Doyle, Rich, Steele & Sussman, *AMORD: A Deductive Procedure System*, MIT AI MEMO, Jan. 1978, at 435. An alternative approach includes ongoing research under the Kripke structures of possible worlds logics at SRI International. *See* S. Rosenschein & L. Kaelbling, The Synthesis of Digital Machines with Provable Epistemic Properties, Draft Report for the Artificial Intelligence Center SRI Int'l and Center for the Study of Language and Info. at Stanford U. (Sept. 18, 1985) (available at Stanford University).

38. There are such techniques, however, and they are starting to be implemented. *See* Manna & Waldinger, *A Deductive Approach to Program Synthesis*, in READINGS IN AI, *supra* note 18, at 141-72; 2 LOGICAL BASIS, *supra* note 31; R. SMULLYAN, FIRST-ORDER LOGIC (1968) (explaining the analytic tableaux method).

39. The "if" operator is interchangeable with the combined "not" and "or" operators,

Resolution and its constraints place significant reliance upon the translating human's ability to capture the context and to comprehend the most probable reasoning approaches.[40] If the program requires Horn clauses, different paths may result from a different form of question, as different predicates come to the front. Maintaining completeness requires a "combinatorial explosion" of either the database or of the control rules, in order to allow for all possible inference paths.

These constraints also make it difficult to express certain concepts. Time and its process (crucial to many aspects of the law or related domains) are both problems. Attitudes, beliefs, or externally imposed concepts, such as morality or necessity, are others. The context of every logical atom (its situation) is yet another representational problem within the constraints acceptable to resolution. Temporal, modal, and situation logics[41] can all be used to handle these problems, but they are

i.e., –HAPPY(Bill) v LUCKY(Bill) is logically equivalent to HAPPY(Bill) → LUCKY(Bill). Furthermore, both "or" and "and" are commutative, so LUCKY(Bob) & HAPPY(Bill) is equivalent to HAPPY(Bill) & LUCKY(Bob). This commutativity and equivalence leads to the combinatorial explosiveness of resolution.

40. A number of different resolution inference strategies exist: unit resolution, where one parent clause at each step must be a single literal; input resolution, where one parent clause at each step must be from the initial data base; linear, or ancestry filtered, resolution, where one parent clause at each step must be from the initial data base or an ancestor of the other parent; lock resolution, where resolution is only permitted on the first literal of each clause; and, set of support resolution, where one parent must be from the clauses resulting from negation of the goal or their decedents. PROLOG typically combines lock, unit, and set of support resolutions. The inference strategy used may or may not preserve completeness, or it may favor speed. *See* LOGICAL FOUNDATIONS, *supra* note 32, at 95.

41. *See* N. NILSSON, PRINCIPLES OF ARTIFICIAL INTELLIGENCE 307-16 (1980) [hereinafter PRINCIPLES OF AI] (recounting some experimental work in the field); G. HUGHES & M. CRESSWELL, AN INTRODUCTION TO MODAL LOGIC (1968) (a good general text; temporal logic is discussed at 262); Kripke, *Semantical Considerations on Modal Logic*, in REFERENCE AND MODALITY 63-72 (1971) (modal logics); Hintikka, *Semantics for Propositional Attitudes*, in PHILOSOPHICAL LOGIC 21 (1969); K. KONOLIGE, A DEDUCTION MODEL OF BELIEF AND ITS LOGICS 326 (SRI Int'l, Artificial Intelligence Center, Tech. Note, 1984) (particular means of application to belief logics) [hereinafter a A DEDUCTION MODEL]; McCarthy, *Ascribing Mental Qualities to Machines*, in PHILOSOPHICAL PERSPECTIVES IN ARTIFICIAL INTELLIGENCE 161 (1979) (a nontechnical paper on the topic).

The problem of computing the context (its changes and, more difficultly, its constants) is known as the "frame problem." *See* McCarthy & Hayes, *Some Philosophical Problems from the Standpoint of Artificial Intelligence*, in 4 MACHINE INTELLIGENCE 463 (1969) [hereinafter *Philosophical Problems*]. Currently, research into whether this can be subsumed as part of the mathematical technique recently proposed by John McCarthy called "circumscription" is being pursued at Stanford, SRI International, and elsewhere. *See* McCarthy, *Circumscription—A Form of Non-Monotonic Reasoning*, 13 ARTIFICIAL INTELLIGENCE 27-39 (1980); J. MCCARTHY, APPLICATIONS OF CIRCUMSCRIPTION TO FORMALIZING COMMON SENSE KNOWLEDGE (1984) (Problems of the Non-Monotonic Reasoning Workshop, sponsored by Am. Ass'n Artificial Intelligence).

imperfectly interwoven, and their exact limitations are currently un-known. In fact, all that is known is that perfection is unattainable: any sufficiently powerful system will be incomplete insofar as unprovable truths exist.[42] The techniques in use, discussed above, are by no means the only mathematics of logic; rather, they represent some of the many alternatives. In order to emulate the flexibility of human reasoning, however, an artificial intelligence, or expert system, program must in-clude all alternative truth tables as well as instructions indicating when each set should be used.[43] Within this environment, what are the un-derlying techniques used to implement reasoning in AI and ES, and what makes these distinguishable from ordinary programs?

C. REASONING WITH HEURISTICS RATHER THAN ALGORITHMS

1. *Algorithms and Heuristics*

Ordinary programs transform their inputs to outputs in a predeter-mined process. The common understanding of a "program" is: a state-ment of the tasks to be performed and the order of their performance. However, two types of processes which answer questions or solve problems, and which reflect distinct underlying models of the world, exist: algorithmic and heuristic reasoning.

An algorithmic process is one whereby "the method of solution must be expressed precisely as a series of independent, clearly-defined

---

42. Since the constraints are mathematical problems, the academic community is pos-itive that some will prove to be insolvable, as they are bound by Gödel's incompleteness theorem. *See* H. ENDERTON, A MATHEMATICAL INTRODUCTION TO LOGIC (1972) (a formal approach); HOFSTADTER, *supra* note 25, at 438 (a more intuitive explanation). The latter reference restates the original theorem in more comprehensible form as: "All consistent axiomatic formulations of number theory include undecidable propositions." *Id.* at 17.

See Gemignani, *supra* note 15, at 183-84. A classic conundrum, incapable of mathe-matical algorithmic solution, is the "halting problem." *See* Turing, *On Computable Num-bers With An Application to the Entscheidungsproblem*, 2 PROC. LONDON MATHEMATICAL SOC'Y 230-65 (1936) (no machine can calculate whether it will halt on all given sets of in-put). *See* HOPCROFT, *supra* note 24, at 146 (describes Turing machines and discusses un-decidability); W. WULF, M. SHAW & P. HILFINGER, FUNDAMENTAL STRUCTURES OF COMPUTER SCIENCE 353 (1981) [hereinafter FUNDAMENTAL STRUCTURES] (description and proof of the halting problem).

43. Of course, it is this innate flexibility that may make the human reasoner less de-pendable. He or she may draw inconsistent and strained analogies to form absurd repre-sentations of the facts, shift from logic to logic heedless of the formal constraints, and use emotional or rhetorical flimflam to conceal the consequential inconsistencies. Our inter-nal truth tables and processes are not subject to as strict a scrutiny as the computer's can be, or to as rigorous a proof of correctness. Some idea of the variations and their differing requirements can be gleaned from Fagin & Halpern, *Belief, Awareness, and Limited Rea-soning: Preliminary Report*, 1985 PROC. NINTH INT'L JOINT CONF. ARTIFICIAL INTELLI-GENCE 491.

steps to be carried out."[44] All algorithmic processes necessarily pre-
sume a perfect domain where all possible states are known, representa-
ble, or attainable by the computer. An algorithmic process can be
evaluated and measured;[45] the computer can calculate either the effi-
ciency of a particular input transformation or the average efficiency of
the entire input domain.[46] To be successful, an algorithmic process
must be able to be performed by utilizing merely those resources avail-
able to the computer. Most computer systems and most problems do
not meet these requirements. Computing time, knowledge, and solution
methods are typically all incomplete. Although an algorithm may be
devisable, the solution may be so expensive, in terms of processing time,
or the value so inexpressibly large, that it is impossible to attain. Algo-
rithms give an illusion of certainty, but they should be considered
flawed until their reliability is actually proven.[47]

In contrast, heuristic processes involve imprecise methods, possibly
interdependent steps, recalculation, and measurement against a goal,
i.e., guessing.[48] Unlike algorithmic processes, heuristic processes work
with imperfect domains. If the heuristic used was well chosen, a great

---

44. *See* Gemignani, *supra* note 15, at 181-83 (a more complete explanation and an ex-
ample of a primitive algorithm).

45. For example, the following BASIC program will run forever:

```
0:     BEGIN
10:    PRINT "HELLO"
20:    GOTO 10
30:    END
```

To transform it into verification language, one must insert an implicit control structure—
"advance to the next line unless otherwise directed, stop only when directed"—but this is
still straightforward. Once 0: becomes true, the precondition of the final statement (ad-
vance to 30:) never becomes true, because the precondition of 20: becomes the precondi-
tion to 10: (advance to 10:). *See* FUNDAMENTAL STRUCTURES, *supra* note 42, at 85.

46. This branch of computer science, known as "analysis of algorithms," measures the
performance feature of algorithmic processes, but does it imprecisely: the functional
mathematics considers $cN = N$ for any value of c or N! What is important is the *power* of
N, i.e., does the algorithm require N^2 repetitions, $2log^N$ repetitions, or $cN^4 + log^N$
repetitions?

47. An example of the first type of problem is the present impossibility of finding the
perfect chess game. An example of the second type of problem is computing values of
Ackermann's function for numbers greater than, say, 20. An example of the third type of
problem is finding a solution (or its contradiction) to Fermat's last theorem ("no solution
for $a^n + b^n = c^n$ exists for any integer value of a, b, c when $n > 2$"). An algorithm to test
this theorem is simple to devise, horribly inefficient, and may never terminate.

48. An example of an algorithm to find the value of 2 is:

```
N = 1;
FOR I = 1 TO 15 DO
     N:N X 2;
PRINT N.
```

An example of a heuristic is:

deal of intermediate work could be eliminated; if the goal were unobtainable, a best possible answer could be reached; if the goal were misstated or poorly understood, a heuristic process could provide vital information that would help correct itself.[49] However, heuristics are also subject to many problems and are harder to evaluate definitively.[50] Reasoning with heuristics involves an attempt to somehow match, devise, or meet a goal condition, or to provide one member of an equivalent class of goal conditions. Unlike an algorithmic attempt to generate every possible state that leads it to end with the goal stated, a heuristic process searches among the reachable states to find a goal state. There is an innate potential for incompleteness that is simultaneously a strength and a weakness.

Algorithms and heuristics have different strengths and weaknesses. Human reasoning includes many different processes, such as mathematical calculation, logical reasoning, information finding, and pattern comparison. Each can be implemented in either an algorithmic or a heuristic fashion; it is not always clear which is better.[51] Algorithms, as well as heuristics, may depend upon information that is unknown when the program starts; either may be interactive. Yet, whenever the task depends upon interaction with the real world, or with the context of the

---

TYPE INTO A LISP INTERPRETER
    (EXPONENT 2 15)
RECEIVE THE REPLY
    UNDEFINED CAR OF FORM — EXPONENT
TYPE IN
    (EXPT 2 15)
RECEIVE THE REPLY
    32768

and accept the latter response.

49. An example of a heuristic is the strategy used in the problem above where the user goes backwards from the (negated) goal sentence and tries to eliminate positive literals at the end of each clause. *See supra* note 32 and accompanying text. Every heuristic must be stated as a rule (or set of rules) and becomes a part of the control strategy for the artificial intelligence or expert system program. If the user simply tried to resolve pairs of sentences in increasing number order, there would be many more false steps.

50. *See* Gaschnig, *A Problem Similarity Approach to Devising Heuristics: First Results*, in READINGS IN AI, *supra* note 18, at 23-29 (examples of evaluation techniques).

51. A trade-off between efficiency and accuracy exists at any level above the individual transfers within the operating system. Because it is defined by the question domain, "Better" involves an analysis of the trade-off. (If the question were, "How many sides does a triangle have?," would "3" be a better answer than "3.0"?) Success also may interact with the process of solution. If an artificial intelligence attempts to convince its examiner that it is actually a human being (one version of the Turing test), the correct way to answer a question may involve irrelevancy or randomness. If the artificial intelligence attempts to teach correct behavior (by learning the typical errors made by beginning psychotherapists and emulating the uncooperative behavior that they produce), the correct way to simulate answers from a paranoid patient may depend on the phraseology of prior questions.

program, to provide this information, the risk that the unknown state (of the world or of the program) will not become known always exists.[52] Under these conditions, algorithms must fail; heuristics may have alternatives to fall back on, alternatives which may depend upon the context of the process as much as the current content of the process. In this sense, heuristics can look outside of their subject matter for clues to solutions, while algorithms are entirely wrapped up in the subject, and, thus, are at the mercy of the subject's problems. Because AI and ES use heuristic reasoning processes, they can be distinguished from ordinary programs. Either may use algorithmic processes as part of their operation, but they contain this additional, adaptable, and amorphous capacity.

2. *Search as the Heuristic Reasoning Method*

For any real world or expert domain, the size of a program plus its database is so great that gaps may be invisible to the human programmer or user. Because these programs, as real world incarnations of mathematical purities, are working with limited resources in computing time and informational accuracy, the programs must include the human function of guessing (a reasonable response when coping with limited resources). The selection and nature of the particular control strategy chosen for the program can influence the results. The heuristics' function (as well as the measurement of their average success) is to maximize the quality of the machine's choice and thus minimize the search space that must be examined.

Typical control structures for algorithmic programs—iteration, single state branching, and multiple state branching—are inadequate for effective control of heuristic programs. This dictates use of what is called "partial programming."[53] Because heuristics may generate wrong answers or lead to incorrect reasoning paths, backtracking methods must be implemented. Also, when feedback from the user or real world may be the source of necessary information, methods must be implemented to allow the new information to alter the control process where

---

52. For example, the World-Wide Military Command System's (WWMCS) computers in Washington, D.C. and Florida work together and are, thus, supposed to stay connected. However, automatic reconnection of the Washington computer failed when a loss of power interrupted communications between the two computers in November 1978, because programmers had not anticipated a need for one computer to sign on twice. Human operators had to find a way to bypass normal operating procedures before they were able to restore communications. *See* Broad, *Compilers and the U.S. Military Don't Mix*, 207 SCIENCE 1183 (1980).

53. *See* LOGICAL FOUNDATIONS, *supra* note 32, at 279, 285 (defines and describes partial programming).

necessary.[54] Some of the feedback or interactive methods include: requiring the new knowledge to start pre-established processes by satisfying a precondition; attaching procedural knowledge to factual objects and depending on internal activity to produce the correct responses; or having one part of the program invoke another upon attaining some generally predicted, but "order-unknown," success in another process.[55]

Finding the right answer (or the best sentence to resolve next, or the best line of reasoning to pursue next)—in essence, the decision of how the program's control process ought to operate from moment to moment—leads to a reasoning problem people usually think of with regard to database or reference finding problems: How can I answer my question?

Nils Nilsson, presently the dean of Stanford University's Computer Science Department,[56] persuasively argued that all artificial intelligence problems can be seen as searching for the right answer.[57] Exhaustive methods exist that could, at least theoretically, examine every possible answer, but these methods take a dramatically increasing amount of time as the range of possibilities expands.[58] For example, in most large problems (difficult domains), the number of logical possibilities is subject to combinatorial explosion. Consequently, the number of logical possibilities will far exceed the capabilities of any foreseeable exhaus-

---

54. These methods can be implemented by algorithmic processes, but there is a crucial difference: an algorithmic process presumes that, at certain stages, the necessary information will become known; a heuristic program is far more opportunistic and must react when the information becomes known, without presuming that it will become known.

55. Alternative control methods for altering the state of a program include object-oriented programming, demons, and blackboards. *See generally* H. ABELSON, G. SUSSMAN & J. SUSSMAN, STRUCTURE AND INTERPRETATION OF COMPUTER PROGRAMS (1985) (object-oriented programming); P. WINSTON, *supra* note 2 (object-oriented programming and demons); R. ROBERTS & I. GOLDSTEIN, THE FRL PRIMER (Artificial Intelligence Lab. at Mich. Inst. Tech., Rep. No. AIM-408, July 1977). *See* 1 HANDBOOK, *supra* note 2 (blackboards used in the HEARSAY speech understanding system); Erman, Hayes-Roth, Lesser & Reddy, *The HEARSAY-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty*, 12 COMPUTING SURVEYS 213-53 (1980) (also in READINGS IN AI, *supra* note 18, at 349-89).

56. Stanford University Computer Science Department has been acknowledged as the premier computer science department in the world for the past two years. The academic and scientific credentials of its present dean are too lengthy to summarize.

57. Nilsson's point is more Socratic than dogmatic; his intent was to challenge the seminar attenders' reasoning. *See generally* N. NILSSON, PROBLEM-SOLVING METHODS IN ARTIFICIAL INTELLIGENCE (1971); PRINCIPLES OF AI, *supra* note 41 (1980); 1 HANDBOOK, *supra* note 2, at ch. 2.

58. Two exhaustive methods are depth-first and breadth-first. Depth-first selects one line of search and pursues it to the end, exhausting all the possibilities beneath this line before moving to the next line of search. Breadth-first examines all choices at each level before going down to the next stage.

tive search methodology.[59] Partial solutions, when attained, may guide the future search. In fact, proven failure, i.e., impossibility, may guide the future search. A wide variety of various search strategy implementations have been designed and analyzed,[60] yet, there is no guarantee of success because the answer (goal) may be unattainable. Therefore, the program may be defeated by the very nature of the domain.

D. TACKLING INCOMPLETE DOMAINS

What are the typical features of the domains where AI and ES cur-
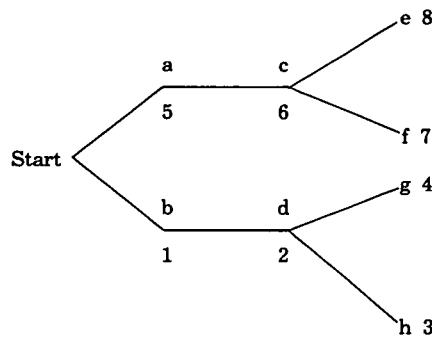
Depth-first follows the numbers; breadth-first follows the letters. The best and worst case results for these are:

	Breadth-First	Depth-First
Best-Case	$$\frac{B^{D-1} - 1}{B - 1} + 1$$	D
Worst-Case	$$\frac{B^{D} - 1}{B - 1}$$	$$\frac{B^{N} - B^{N-D}}{B - 1}$$

B = Branching Factor
D = Depth of Answer
N = Max. Depth

59. In theory, every possible move of every possible chess game could be laid out in a computer's memory, and it could search for the right move each time it moved. However, since the total number of moves in the average game has been estimated at 10^{120}, the search time is far too great to be possible, let alone practical. The size of the search space for checkers is barely attainable, but still large—10^{50}. *See* 1 HANDBOOK, *supra* note 2, ch. 2, at 27.

60. A partial listing of various strategies includes: Hill-climbing, Beam, Best-first, British Museum, Branch and Bound, Dynamic Pruning, and Heuristic Continuation. *See generally* 2 HANDBOOK, *supra* note 2; P. WINSTON, *supra* note 2; AI J.: SPECIAL AI JOURNAL ISSUE ON SEARCH AND HEURISTICS, Mar. 1983.

rently operate, and when are heuristic programs better suited than algorithmic programs?

1. *Incomplete Domains*

Certain problems will never be amenable to mathematical analysis and thus require heuristic methods to guide the search for an acceptable solution.[61] Other domains are incompletely understood, either because they have not been perfectly specified yet, or because they are inherently open-ended domains. For open-ended domains, the boundaries of the search space are not readily distinguishable,[62] and complete delimitation is inherently impossible. As long as these limitations exist, such domains are inherently beyond present or future capabilities to ensure algorithmic perfection. Thus, heuristic methods are the appropriate approach.

2. *Imperfect Domains*

A second class of problem domains are those where there is imperfect[63] knowledge about the domain. For example, entering all the truths before a search may not be practicable. Alternatively, areas of uncertainty may exist, such as uncertainty about the data, or uncertainty about the correctness of inferences.[64] Apart from epistemological considerations, uncertainty in the data can be imposed by linguistic constraints,[65] by semantic constraints,[66] or by the input itself, if permit-

---

61. A large number of problems are "NP-Complete" (nondeterministic polynomial-time), i.e., for all practical purposes their solution by an algorithmic process on any computer cannot be guaranteed. One such problem is the travelling salesman/Hamilton circuit. This problem asks whether, for any route/graph, there is a path from the start/origin whereby each city/point is visited exactly once before returning to the start/origin. "NP-Hard" problems pose pragmatic limitations: in theory, computing 2^6 and 2^{7637} are not very different, but, in practice, the second calculation is much, much harder. *See* HOPCROFT, *supra* note 24, at 320.

62. For example, consider comprehension of pronunciation and mispronunciation ("I scream" vs. "eyes cream" vs. "ice cream") and the area of natural (as opposed to mathematical) language processing. *See* 1 HANDBOOK, *supra* note 2, at ch. 4, 5; T. WINOGRAD, LANGUAGE AS A COGNITIVE PROCESS (1983); Winograd, *What Does It Mean To Understand Language?*, 4 COGNITIVE SCI. 209-41 (1980). For a good critique of the expert system approach, see T. WINOGRAD & F. FLORES, UNDERSTANDING COMPUTERS AND COGNITION (1986).

63. One meaning of "imperfect" is not legally enforceable. *See* AMERICAN HERITAGE DICTIONARY (4th ed. 1969).

64. This amount of uncertainty should be familiar, as it is no greater than the practiced response of any attorney given the opportunity to qualify his conclusions.

65. In *Frigaliment Importing Co. v. BNS International Sales Corp.*, 190 F. Supp. 116, 117 (S.D.N.Y. 1960), Judge Friendly stated, "The issue is, what is chicken?" The standard legal technique of redefining imperfectly stated concepts to obtain a solution was considered (by analogy) in Amarel, *On Representation of Problems of Reasoning About Actions,*

ted.[67] Uncertainty in the rules can be imposed by using inference keyed to probability calculations,[68] evidential reasoning,[69] fuzzy logics,[70] probabilistic entailment,[71] or the above-mentioned certainty factors.

Arbitrary means may be employed to complete the knowledge base used by the AI or ES, or to provide the ability to make assumptions and to retract them upon receipt of contrary evidence. These means are just as heuristic as any other means employed to render the problem soluble

---

in READINGS IN AI, *supra* note 18, at 2-22. If an input comes from a thermometer that is hovering between 99°C and 100°C and the computer must act when boiling temperature is reached, how should it handle this uncertainty?

66. For example, how can a program represent the concept "unknown"? *See* Patel-Schneider, *A Decidable First-Order Logic for Knowledge Representation*, 1985 PROC. NINTH INT'L JOINT CONF. ARTIFICIAL INTELLIGENCE 455-58. *See* Hayes, *Some Problems and Non-Problems in Representation Theory*, in READINGS IN KR, *supra* note 2, at 3-22 (an overview). In *City of Lincoln v. Bud Moore, Inc.*, 210 Neb. 647, 316 N.W.2d 590 (1982), the basic problem arose when the utility company's computer could not represent a five-digit value on a four-digit electrical meter.

67. For example, the medical diagnosis programs developed around EMYCIN operate with certainty factors; the doctor can include in the answer a qualification of how certain he or she is that it is true (e.g., "yes (.8)"). *Cf.* 2 HANDBOOK, *supra* note 2, at 184-92; E. SHORTLIFFE, COMPUTER-BASED MEDICAL CONSULTATIONS: MYCIN (1976); B. BUCHANAN & E. SHORTLIFFE, RULE-BASED EXPERT SYSTEMS: THE MYCIN EXPERIMENTS OF THE STANFORD HEURISTIC PROGRAMMING PROJECT (1984) [hereinafter MYCIN].

68. *See generally* LOGICAL FOUNDATIONS, *supra* note 32, at 177. Straight probability calculations involve Bayes' rule:

$$p \ (Q \mid P) \ = \ \frac{p \ (P \mid Q) \ p \ (Q)}{p \ (P)}$$

(the conditional probability of Q given the prior probabilities of P, Q, and their relationship). This approach obviously becomes combinatorially, computationally expensive and requires knowledge of the prior probabilities, although approaches that estimate or use conditional probabilities have been explored.

69. Evidential reasoning was suggested in Dempster, *A Generalization of Bayesian Inference*, 30 J. ROYAL STATISTICAL SOC'Y 205-47 (Series B, 1968), and extended in G. SHAFER, A MATHEMATICAL THEORY OF EVIDENCE (1976). The Dempster-Shafer theory of evidence was recently examined with regard to possible use in the MYCIN system in Gordon & Shortliffe, *A Method for Managing Evidential Reasoning in a Hierarchical Hypothesis Space*, 26 ARTIFICIAL INTELLIGENCE 323-57 (1985).

70. Fuzzy logic was studied in Zadeh, *Fuzzy Logic and Approximate Reasoning*, 30 SYNTHESE 407-28 (1975), and is the basis of the language FUZZY developed at Rutgers University. *Cf.* R. LeFAIVER, FUZZY REFERENCE MANUAL (1977) (available at the Computer Sci. Dept., Rutgers Univ.).

71. Probabilistic entailment is discussed in LOGICAL FOUNDATIONS, *supra* note 32, at 193. It involves modifying the standard truth table approach for modus ponens (or the appropriate inference rule used by the program), to include consistent truth value assignments as a matrix value. The reasoning then involves matrix mathematics and results in bounded probability assessments. At present, this has not been incorporated in any system and is subject to size considerations.

by the computer, and they are the topic of current research.[72] These approaches all attempt to deal with a realistic reflection of the uncertainty that exists in the real world.

3. *Modelling Domains*

Any AI or ES must model its domain because that domain will be composed, in part, of outputs produced by the program.[73] The modelling can be implicit in the function of the program, or the modelling can attempt an explicit, evaluative, internal reference functionality (i.e., self-consciousness).

Any interactive modelling that includes the real world, i.e., any AI, necessarily uses reason rather than logic. The difference between logic and reason is that the former is based on mathematics and the latter is based on the real world. Therefore, logic can be tested and evaluated and can produce conclusions within its own internal frame—all measured in terms of consistency (subject of course to the problem identified in *Gödel, Escher, Bach: An Eternal Golden Braid*[74]). The latter can be tested and evaluated and can produce its own conclusions only in reference to the externalities surrounding it—all measured in terms of "truth" (defined as the correspondence with events). Any modelling that includes an imperfect domain, if evaluated outside that domain, also uses reason to establish the constraints, heuristics, and evaluative techniques that provided the operative model.

The domain-model can be composed of objects, processes, actions, or situations. The computer, its user, and possible other parties, can be formulated as one or more agents, who may have properties expressed as intentions,[75] knowledge or beliefs,[76] or stances.[77] The richer the

---

72. *See* Reiter, *On Reasoning By Default*, in READINGS IN KR, *supra* note 2, at 401-10; Hayes, *The Second Naive Physics Manifesto*, in READINGS IN KR, *supra* note 2, at 467-86; Moore, *Reasoning About Knowledge and Action*, in READINGS IN AI, *supra* note 18, at 473-77.

73. One of the major discoveries of the MYCIN project was the necessity of an explanation function that can trace from the conclusion to the intermediate steps, the rules used, and the inputs from the outside world that led to that conclusion. *See generally* MYCIN, *supra* note 67.

74. D. HOFSTADTER, *supra* note 25.

75. *Cf.* Maida, *Knowing Intentional Individuals, and Reasoning About Knowing Intentional Individuals*, 1983 PROC. EIGHTH INT'L JOINT CONF. ARTIFICIAL INTELLIGENCE 382-84.

76. *See* A DEDUCTION MODEL, *supra* note 41, at 3.

77. A "stance" is a hierarchically-organized, pre-set description of a situation with the possessor's reactions already defined. Situation-specific details are incorporated into a selected stance to provide situation-specific activity. (An example is a thoroughly hostile (or bored) judge replying, "Overruled" to every shout of "Objection" by counsel). This categorization of activity can provide immediate or limited resources when complete processing for the perfect reaction is neither possible nor cost-efficient. John K. Myers, at SRI

model, however, the more likely the program will be required to balance the problems of combinations and their tradeoff—the risk of non-coverage through incomplete heuristics. The very richness of detail can make problem-solving a great deal more difficult, if not beyond the limited resources available.[78]

A part of the modelling process, which, in effect, is the machine's consciousness of its surroundings and its processes, involves the awareness, or more often, the lack thereof, of the meaning of the inputs. Since this is where human experts, and even laymen, excel at their average performance, it is odd that many modern systems lack this awareness. There are two reasons for this deficiency:

> (1) Incomplete specification of the domain at all possible levels renders the concept of possible range intransigent to realization. This is in part why handwriting and even printed text is difficult for a machine to process despite knowledge of individual characters out of context;[79] it is potentially the core problem for machine vision.[80]
>
> (2) The system may presume dependance upon the human user to provide judgment and act as a filter because of the limited set of persons to whom a system is released.[81] In effect, there is a default assumption of the possible contexts, however, specifying all that is excluded is an open-ended problem.[82]

Any program operating at all in real time, or involving even a subset of the open-ended and infinitely-variable real world (such as an air-

---

International, proposed that stances should be used autonomous agents (e.g., robot exploration vehicles).

78. *See* 1 HANDBOOK, *supra* note 2, at ch. 2; Amarel, *On Representations of Problems of Reasoning and Actions,* in READINGS IN AI, *supra* note 18.

79. *See* AMERICAN NATIONAL STANDARDS INSTITUTE, INC., DRAFT PROPOSED AMERICAN NATIONAL STANDARD GRAPHICAL KERNAL SYSTEM (1984); Ward & Blesser, *Interactive Recognition of Handprinted Characters for Computer Input,* IEEE COMPUTER GRAPHICS & APPLICATIONS, Sept. 1985, at 23, 24.

80. *Cf.* 3 HANDBOOK, *supra* note 2, at ch. 13; P. WINSTON, *supra* note 2, at 335, 341; Brady, *Computational Approaches to Image Understanding,* COMPUTING SURVEYS, Mar. 1982, at 3. *See* Chin & Harlow, *Automatic Visual Inspection: A Survey,* in IEEE TRANSACTIONS OF PATTERN ANALYSIS & MACHINE INTELLIGENCE 557 (1982) (general sense of applications in industry).

81. MYCIN began as a research project and was never intended for general use; a close descendant, PUFF (which diagnoses pulmonary function) was released only to interested doctors at San Francisco City Hospital. *See* 2 HANDBOOK, *supra* note 2; STANFORD COMPUTER SCIENCE DEP'T, HEURISTIC PROGRAMMING PROJECT REPORT NO. HPP-78-19, A PHYSIOLOGICAL RULE-BASED SYSTEM FOR INTERPRETING PULMONARY FUNCTION TEST RESULTS (1978).

82. Default reasoning as an approach to common sense must be limited to a particular domain. A temperature of 98.6+o is a reasonable default if measured in Fahrenheit in a human patient, but not if measured in Celsius in a house. Default reasoning and other approaches to what is called "nonmonotonic reasoning" are under investigation. *See* LOGICAL FOUNDATIONS, *supra* note 32, at 115; Reiter, *A Logic For Default Reasoning,* 13 ARTIFICIAL INTELLIGENCE 81-132 (1980).

traffic controller, steel-smelter monitor, or nuclear power plant control-
ler),[83] must sense (i.e., model from an input such as a gauge reading to
temperature) what is occurring in that world and provide correspon-
dences between its reasoning, and actions and effects in the real
world.[84] Furthermore, it must adapt to changing knowledge, either as
the world changes or as the knowledge about the world grows. The
human ability to adapt to changing circumstances, or changing truths, is
not easily duplicated.[85] As part of the environment, the AI or ES needs
some form of feedback.

Learning is currently a very active research area.[86] Researchers
have tested the partially automated interactive transfer of expertise ap-
proach to learning. Randall Davis worked with the MYCIN team to get
past the knowledge engineering bottleneck and developed the TEREI-
SIAS program; he suggests a possible improvement to an investment ad-
visor ES as an example.[87] The incompleteness or presumption of
perfection in the program, the knowledge base, or the model of the real
world and all possible interactions, may exist as unspoken and unac-
knowledged assumptions about the world.[88] At this point, this is simply
worth noting as an issue of concern.[89] These assumptions begin to lead

---

83. *Cf.* Nycum, *Liability for Malfunction of a Computer Program*, 7 RUTGERS COM-
PUTER & TECH. L.J. 1 (1979).

84. "A computer program capable of acting intelligently in the world must have a
general representation of the world in terms of which its inputs are interpreted. . . . The
epistemological part [of intelligence] is the representation of the world in such a form that
the solution of problems follows from the facts expressed in the representation." *Philo-
sophical Problems*, *supra* note 41, at 431.

85. If there was a default rule that anyone who was not provably rich was not rich,
there would be a conflict. Earlier, -RICH(Sam) was deduced; now the default rule would
produce -RICH(Sam). Rule (3) was eliminated, but not the deduction that depended
upon it. Default logics allow assumptions to be used, but their knowledge base can
shrink as well as grow, i.e., they are "nonmonotonic."

86. *See generally* 3 HANDBOOK, *supra* note 2, at ch. 14; Lenat, *The Role of Heuristics
in Learning By Discovery*, in MACHINE LEARNING: AN ARTIFICIAL INTELLIGENCE AP-
PROACH 243 (1983); T. MITCHELL, L. STEINBERG & S. AMAREL, A LEARNING APPRENTICE
FOR KNOWLEDGE ACQUISITION IN AN EXPERT SYSTEM 16 (Rutgers Digital Design Project
Working Paper, 1984).

87. *See* Davis, *Interactive Transfer of Expertise: Acquisition of New Inference Rules*,
in READINGS IN AI, *supra* note 18, at 410-28. These latter functions usually are available
only when the range of responses is relatively predictable (such as a text editor receiving
a non-character code over the modem), or the means of transferring into program-com-
prehensible terms is provided and the process can be tested by the expert. Then the judg-
ment can be shared or the risk can be shared between the human and the computer. *Id.*

88. One prototypical example is reasoning about employees in a company, a set whose
elements change constantly. Another is defining for a factory-control AI application, the
location of all of its subordinate machinery, and then coping with layout changes or tem-
porary dislocations required for repairs. A third is updating the location, headings,
speeds, fuel status, and destinations of airliners for an air traffic control program.

89. For example, consider the assumption that a payroll program contains every com-

to the problem of what the AI or ES is expected to do, what users it is expected to have, or, in short, what contacts the rest of the world will have with the program.

E.   JURISDICTION, VENUE, AND CIRCUMSCRIPTION
OF THE QUESTION DOMAIN

As stated in the preface, this Article assumes that some individual will allege that the new technology injured him. That injury could occur in a number of ways, and how that injury occurs will affect the appropriateness of the legal theories considered.[90]

One analyst suggested that courts use the following test to determine the rights of computer programs; the same test might be used to assign responsibilities:

> The most logical test would be the one used for humans: behavior. If the program proved capable of reasoning within the limits imposed by society, the computer would have proven itself competent. Since, at the present time, programs are specialized, the computer should presumably be considered competent only in the area of specialization.[91]

However, this presumes that the AI or ES itself accepts liability. This premise is beyond the scope of this Article if only because presently all AI and ES programs are judgment-proof.[92]

---

pany employee in its employee database; this is sufficient until the tax program asks about consultants' fees. *See* Reiter, *On Closed World Data Bases*, in READINGS IN AI, *supra* note 18, at 119-40.

90. At least two firms specialize in the adolescent area known as "knowledge engineering": Teknowledge, Inc. of Palo Alto, California, and Intellicorp, Inc. of Mountain View, California. *Cf.* BUILDING ES, *supra* note 2, at xv. (The author apologizes for a West Coast locality bias, it is hard to keep up with this extraordinarily fast-moving field, and the author recognizes other centers of interest and development include Boston, the Texas Gulf, Tokyo, London, and Paris.) These firms provide their corporate customers with the support software for building expert systems (including specialized graphics displays and application languages: S1 and M1 for Teknowledge, KEE for Intellicorp), knowledge engineering design, and training personnel and courses. A new wave of production rule software or expert system shells for build-your-own expert systems is also in the marketplace, with prices ranging from $75 to $15,000. Anyone can study the literature, learn the programming language of their choice (LISP, the traditional AI language, has been challenged by C. Pascal, and a host of company-specific languages or dialects), and build (or more likely attain partial success in building) an artificially intelligent program. The field is changing on a monthly basis, with major advances practically stumbling atop each other.

91. Willick, *AI: Some Legal Approaches and Implications*, AI MAG., Summer 1983, at 5-16.

92. If one considers that the AI or ES is an entity on its own, other causes of action open up (e.g., respondeat superior). Whether these are appropriate will depend both on the circumstances and on the open-mindedness of the legal forum. For the foreseeable future, programs probably will be objects and not persons. In the near future, artificial intelligence probably neither will be granted legal status as an independent or even semi-

Once an injury occurs, however, the injured party will presumably seek legal recompense. Some of the legal background assumptions will be laid out below but with far less attention to detail than the technical background assumptions behind the program, because the readers are presumed to be more familiar with these topics.

A basic assumption is that the potential for injury extends across society, from powerful corporations to individuals, to both direct and indirect users. Modern American Joneses have access—whether directly (by merely purchasing a personal computer), or indirectly (by accessing expansive databases or interactive bulletin board systems via modem)—to more memory, processing power, and interactive ability, than the military had in the early '60s. Program availability is rapidly moving downward in the marketplace[93] such that this new technology should be considered more analogous to the automobile or the home television (which as a video monitor, may be part of the family computing system) than to railroads, nuclear power, communications satellites, or other large-scale technologies that have altered the shape and nature of our society.

Furthermore, this Article assumes that courts will exercise jurisdiction over defendants responsible for these types of injuries. The spread of electronic technology, and the ability of the computer program to reach into or out of every home across the world, makes this a reasonable extrapolation. The transmission and reception of electrons mark-

---

independent entity, nor will an advance in the field occur that both would impact the body public and make its way up to the appellate level. The technical problems in creating any truly independent artificial intelligence, one that is capable of learning, growth, change, consciousness and self-consciousness (as opposed to knowledge about its self and structure), appear to be recalcitrant to any near-term solution. Perhaps after forty more years of experimentation and practice, there may be a truly independent artificial intelligence, such that science fiction becomes reality, again. Before that time, however, some of the possible approaches to legal liability which do not require that the program be accorded any distinct status as a legal entity can be explored.

For those interested in the issues that may be involved in according a separate legal status to artificial intelligence, see Willick, *supra* note 91, at 7-11; Wilks, *Responsible Computers*, 1985 PROC. NINTH INT'L JOINT CONF. ARTIFICIAL INTELLIGENCE 1279-80. Willick suggests that a slave rationale may be apropos.

For prospective ethical reasons, this author finds the legal concept of slavery, even for computers, to be repugnant. The social unacceptability of the concept of slavery persuaded the author to forego further pursuing the line suggested by Willick (with all due respect) and to leave off research and review of possibly pertinent, although dated, precedents. There is nothing in the law to prevent AI or ES services, *qua* services, being owned (the ban is against people being owned) and there would be little or no theoretical paradox in allowing ownership of AI or ES, absent a determination of personhood.

93. In fact, the market may be approaching the point akin to most software where "two programmers and a dog could start a new software house." Rumbelow, *Liability for Programming Errors*, 9 INT'L BUS. L. vii-viii (1981) (surveying comparative legal approaches). *Cf. infra* text accompanying note 269.

ing interaction with the AI or ES program need be no farther away than the ubiquitous telephone.

Tracing the injury through this electronic web might involve the same travails of proof as tracing the paper trail of manufacturing and distributing physical items. The injured party's attorney will have to understand the defendant's ability to mislay, lose, or fog the traces granted with electronic editing and file-changing forgery. These issues of proof and evidence are subsumed into the assumption that if the AI or ES program injured a party, that party may find a proper forum to pursue his remedy as well as the means to prove the factual details that support (or fail to support) his claim.[94]

F. INVOCATION OF DUTY: TORT CAUSES OF ACTION

Having found the forum and the proof, what claims, besides those in contract, should the injured party consider?[95] The law intervenes

---

94. For an example of the choice of law and venue issues that may be involved in information liability cases, see *Saloomey v. Jeppesen & Co.*, 707 F.2d 671 (2d Cir. 1983).

95. Many cases and articles are concerned with issues under contract causes of action (including the availability or limits of direct and implied warranties, limitations of liability, the applicability of the Uniform Commercial Code, and whether a computer program can be considered a good as opposed to a service). Other cases and articles discuss actions based on fraud or negligent misrepresentation. These causes of action are not drastically different when AI and ES, rather than algorithmic programs are involved; hence, they are not discussed in this Article.

Cases discussing express and implied warranties and the effect, or lack thereof, of warranty limitations include: Southwest Forest Industries, Inc. v. Westinghouse Electric Corp., 422 F.2d 1010, 1013 (9th Cir. 1970), *cert. denied*, 400 U.S. 902 (1970); Sperry Rand Corp. v. Industrial Supply Corp., 337 F.2d 363 (5th Cir. 1964); Kalil Bottling Co. v. Burroughs Corp., 127 Ariz. 278, 619 P.2d 1055 (1980); Salt River Project Agricultural Improvement and Power District v. Westinghouse Electric Corp., 143 Ariz. 437, 694 P.2d 267 (1983), *approved in part, vacated in part*, 143 Ariz. 368, 694 P.2d 198 (1984); Bakal v. Burroughs, 74 Misc. 2d 202, 343 N.Y.S.2d 541 (1972); and W.R. Weaver Co. v. Burroughs Corp., 580 S.W.2d 76 (Tex. Civ. App. 1979).

One case upholds contractual limitations on liability for contract action: Farris Eng'g Corp. v. Service Bureau Corp., 276 F. Supp. 643 (D.N.J. 1967), *aff'd*, 406 F.2d 519 (3d Cir. 1969),

Cases concerning the nature of computer programs (or tapes and other media containing them) include: Computer Servicenters, Inc. v. Beacon Mfg. Co., 328 F. Supp. 653 (D.S.C. 1970), *aff'd*, 443 F.2d 906 (4th Cir. 1971) (computer services not a good under the U.C.C.); Honeywell Information Services v. Demographic Systems, Inc., 396 F. Supp. 273 (S.D.N.Y. 1975) (considered goods and services for purposes of replevin); and F & M Schaefer Corp. v. Electronic Data Systems Corp., 430 F. Supp. 988 (S.D.N.Y. 1977), *aff'd*, 614 F.2d 1286 (2d Cir. 1979) (computer program held to be a good to avoid the statute of limitations).

Cases granting recovery under fraud or negligent misrepresentation causes of action include: Convoy Corp. v. Sperry Rand Corp., 601 F.2d 385 (9th Cir. 1979), *appeal after remand*, 672 F.2d 781 (9th Cir. 1982); and Clements Auto Co. v. Service Bureau Corp., 298 F. Supp. 115 (Minn. 1969), *aff'd in part, rev'd in part*, 444 F.2d 169 (8th Cir. 1971).

when a duty to render justice is invoked. Regardless of the particular theory of justice used to support legal intervention,[96] it is wise to consider how the law determines duty when evaluating a possible cause of action (the form of invocation as sanctified by precedent or reasoned argument). In *Suter v. San Angelo Foundry & Machine Co.*,[97] the court found that:

> The determination of the issue of duty and whether it includes the particular risk imposed on the victim ultimately rests upon broad policies which underlie the law. These policies may be characterized generally as morality, the economic good of the group, practical administration of the law, justice as between the parties and other considerations relative to the environment out of which the case arose. They are found in all decisions whether based on former decisions of the court or on a fresh consideration of the factors found in the current environment.[98]

Most articles, to date, have considered basic computer programs[99] or have been limited to a single cause of action.[100] However, there may be many reasons to claim a tort cause of action. For example, a tort cause of action may be claimed where: (1) the essence of the injury lies in tort, (2) there is a need to avoid a statute of limitations, or (3) there is a concern that the court will uphold a "lack of privity" defense.[101]

---

96. *Compare* Fletcher, *Fairness and Utility in Tort Theory*, 85 HARV. L. REV. 537 (1972) *with* Calabresi & Melamed, *Property Rules, Liability Rules, and Inalienability: One View of the Cathedral*, 85 HARV. L. REV. 1089 (1972) [hereinafter Calabresi] *and* H.L.A. HART, THE CONCEPT OF LAW 151-81 (1961) ("Justice and Morality" chapter).

97. 81 N.J. 150, 406 A.2d 140 (1979).

98. Id. at 173, 406 A.2d at 151 (quoting Green, *Duties, Risks, Causation Doctrines*, 41 TEX. L. REV. 42, 45 (1962)).

99. For a review of the possible approaches available for mass-marketed applications software, and a plea for application in those situations of products liability via uniform nationwide legislation, see Schneider, *Taking the "Byte" Out of Warranty Disclaimers*, 5 COMPUTER/L.J. 531 (1985), and *New England Watch Corp. v. Honeywell, Inc.*, 11 Mass. App. Ct. 948, 416 N.E.2d 1010 (1981) where the court held that a burglar alarm was not a consumer good. This raised a doubt as to whether AI or ES would be considered a consumer good. For a very early paper on the topic see Jordan, *The Tortious Computer: When Does EDP Become Errant Data Processing?*, 4 COMPUTER L. SERV. REP. 1 (1972).

100. For articles discussing products liability, see Gemignani, *supra* note 15; Freed, *Products Liability in the Computer Age*, 16 JURIMETRICS J. 270 (1977); Hall, *Strict Products Liability and Computer Software: Caveat Vendor*, 4 COMPUTER/L.J. 373 (1983); Walker, *Computer and High Technology Product Liability in the 1980s*, 19 FORUM 684 (1984); Note, *Negligence: Liability for Defective Software*, 33 OKLA. L. REV. 439 (1980) (authored by Jim Prince) [hereinafter *Negligence*]; Comment, *Computer Software and Strict Products Liability*, 20 SAN DIEGO L. REV. 439 (1983) (authored by Susan Lanoue). For an article discussing malpractice, see Note, *Disclosure and the Computer Programmer*, 5 COMPUTER/L.J. 557 (1985) (authored by Sam Kraemer) [hereinafter *Disclosure*] (includes discussion of contract and quasi-contract causes of action). For concerns of the advising attorney see Mathew, *Architects, Engineers, Computer Product and the Law: A Matter of Anticipation*, 3 COMPUTER/L.J. 337 (1982).

101. *See Salt River Project*, 143 Ariz. at 437, 694 P.2d at 267. Where a buyer sued for

Available tort causes of action fall into four categories: products liability, services liability, malpractice, and negligence. Each will be examined below.

This is a theoretical Article; there is no known decision involving any AI or ES program. After examining each theory, this Article examines four hypothetical cases (two AI, two ES) in an effort to demonstrate when and why a theory is, or is not, applicable. Brief descriptions of these hypothetical cases follow.

The first hypothetical involves an injury at a chemical factory. In this hypothetical, defendant develops an automated (AI) chemical factory. The system performs a majority of operations without human supervision, including: materials transportation, process supervision, and standardized maintenance. The scope and size of the AI includes knowledge about the chemical reactions currently involved in plant production, but this knowledge can be changed for new or different processes as demands shift. To gain this flexibility, a certain amount of efficiency is sacrificed because lead time is required for non-operation reasoning.

Subordinate AI interactions include: the vision-and-movement activity of the maintenance crew, the materials-handling crew, and the processing crew. If a hazardous situation arises, the system sends signals to human supervisory staff, who will either intervene directly (via remotes) or call for a shutdown that will permit on-the-spot human intervention. The latter option has an inherent inescapable time delay, so the system contains a provision for emergency shutdown and dumping, if the time delay gives rise to disasters.

The second hypothetical involves an injury in a home from an active environmental monitor (AI). In this hypothetical, defendant develops a home environmental monitor that can be sold to any consumer. If the system is installed according to the developer's instructions, the monitor performs the following functions: (1) maintains a user-specified schedule of temperatures (using heating or air-conditioning as required) and lighting; (2) warns of fires and invokes sprinkler systems (optional); (3) optimizes electrical consumption by delayed operation of energy-intensive devices (such as washers, dryers, dishwashers, or water-heaters) to take advantage of non-peak hours; (4) watches for burglaries involv-

economic damages, the Kansas Supreme Court concluded that "implied warranties of fitness and merchantability are not extended to a remote seller or manufacturer of an allegedly defective product, which is not inherently dangerous, for only economic loss suffered by a buyer who is not in contractual privity with the remote seller or manufacturer." Professional Lens Plan Inc. v. Polaris Leasing Corp., 234 Kan. 742, 675 P.2d 887, 898-99 (1984). In a later and related case the court affirmed its position, further finding no agency relation between manufacturer and dealer. See Professional Lens Plan, Inc. v. Polaris Leasing Corp., 238 Kan. 384, 710 P.2d 1297 (1985).

ing breaking and entering; (5) tracks special medical signals from pace-makers or other sensors and rings for emergency medical assistance if a threshold is exceeded (optional); and (6) makes preprogrammed emergency calls in response to special user-selected activation codewords.

The third hypothetical involves an injury from use of pharmaceuticals interaction advisor (ES). In this hypothetical, defendant develops an ES and makes it available to any pharmacist who chooses to use it. The ES uses a correct and complete inference system, includes every legal pharmaceutical in the knowledge base, and its rules include every known interaction. The plaintiff is a pharmacist who seeks indemnity for a malpractice award; the plaintiff in the underlying suit suffered physical injury as the result of taking a prescription that was given to the plaintiff based on the ES's advice.

The fourth, and final, hypothetical involves an injury from use of a financial planning advisor (ES). In this hypothetical, defendant develops an ES and makes it available to the average personal income tax-payer. The ES does not pretend to be complete. In fact, it contains multiple limitation warnings, and includes, as part of its rules, a discussion of its ability to derive inferences, thereby suggesting to the user that human expertise or consideration might be necessary. The system contains a limited portion of the tax law (at a set date) in the form of production rules, but it does not include the capacity to interpret and suggest potential results. The program presumes that determinations are for U.S. valuations and decisions, that dollar sums are involved, and that dollar sums are to be rounded-off to two decimal places. The plaintiff is a purchaser who suffered personal economic loss.[102] These four hypotheticals will be used to provide the factors required by the court in *Suter*,[103] as discussed above.[104]

III.   PRODUCTS LIABILITY: PARTIALLY APPLICABLE AT BEST

A.   THE *RESTATEMENT* DEFINITION AS A JURISDICTIONALLY-NEUTRAL STARTING POINT

The definition of "products liability" in the *Restatement (Second) of*

---

102. The events and nature of injury will be altered each time. For an example of how this sort of reasoning approach might be modelled, see Ashley, *Reasoning By Analogy: A Survey of Selected AI Research With Implications for Legal Expert Systems*, in COMPUTING POWER AND LEGAL REASONING 105 (C. Walker ed. 1985) [hereinafter COMPUTING POWER], and Rissland, *Argument Moves and Hypotheticals*, in COMPUTING POWER, *supra*, at 129.

103. 81 N.J. 150, 406 A.2d 140 (1979).

104. *See supra* text accompanying note 98.

Torts is used as a jurisdictionally-neutral starting point in examining this cause of action.

> Special Liability of Seller of Product for Physical Harm to User or Consumer
>
> (1) One who sells any product in a defective condition unreasonably dangerous to the user or consumer or to his property is subject to liability for physical harm thereby caused to the ultimate user or consumer, or to his property, if
>
> (a) the seller is engaged in the business of selling such a product, and
>
> (b) it is expected to and does reach the user or consumer without substantial change in the condition in which it is sold.
>
> (2) The rule stated in subsection (1) applies although
>
> (a) the seller has exercised all possible care in the preparation and sale of his product, and
>
> (b) the user or consumer has not bought the product from or entered into any contractual relation with the seller.[105]

The basic elements of this cause of action for an AI or ES application thus become: (1) the AI or ES must be a "product"; (2) the defendant must be a seller of the AI or ES; (3) the AI or ES must reach the injured party without substantive alteration; (4) the AI or ES must be defective; and (5) the defect must be the source of the injury. Each of these elements is examined in turn.

B. THE ELEMENTS AS APPLIED TO AI OR ES

1. *When is the AI or ES a Product?*

In the foreseeable future, no general AI product will be roaming about in the real world, either as a mobile robot or as a freely-circulating program.[106] Inevitably, specific environment AI applications and domain-specific ES applications will cause some related economic losses, property or personal injuries, or deaths.[107] Pragmatic limitations on knowledge implementations and commercial conditions ensure that the AI or ES will be created and sold to perform within a stated, limited, environment. Gradually, mass-marketed applications will appear which will be sold to the world at large, and not just on an individually specified basis. For a products liability cause of action to apply, the AI or ES

---

105. RESTATEMENT (SECOND) OF TORTS § 402A (1964).

106. SRI International is programming a mobile robot—the successor to "Shakey," affectionately known as "Flakey"—to research operations in real time in constrained environments, such as building corridors, parking lots and driveways. Independent robots are still the subject matter of science fiction. *See, e.g.*, W. GIBSON, NEUROMANCER (1984); R. ZELAZNY, MY NAME IS LEGION 141 (1976) (story entitled: *Home is the Hangman*).

107. *See infra* text accompanying note 324.

must be a "product";[108] accordingly, two criteria must be met. First, a product—as distinguished from a sevice—must consist of some physical embodiment that is available to the purchaser directly.[109] Second, the AI and ES must not be a unique or specially designed item.

a. "Product" must be interpreted within a defined environment.

Case law has advanced the definition of "product" to the point where the physical embodiment may be very abstract indeed. If water[110] and electricity[111] are considered to be products, why not these applications programs? Whether an injury is caused by an action taken by the AI or ES, an output of the AI or ES, or a particular justification offered by the ES for its reasoning, is arguably irrelevant as long as the AI or ES may be considered an "object." In *Fluor Corp. v. Jeppesen*,[112] following *Lowrie v. City of Evanston*,[113] the court looked not to the dictionary meaning of "product," but rather "within the meaning of its use."[114]

A fine distinction still must be drawn between the output of the AI or ES, independent of its environmental context (i.e., the context-free information produced), and the output as it is interpreted within the environment. It is the output that establishes "the meaning of its use."[115] The need to consider the environment in which the AI or ES operates, as well as the result generated by the program, is highlighted by the different outcomes in two usury cases in Arkansas. In *Cagle v. Boyle Mort-*

---

108. *Cf.* Brannigan & Dayhoff, *Liability for Personal Injuries Caused by Defective Medical Computer Programs*, 7 AM. J.L. & MED. 123 (1981) [hereinafter *Defective Programs*]. "A second criteria for distinguishing services from products is the concept of ownership. . . . [O]wnership is a powerful tool in determining if and when a program has become a product. Similarly, it provides a focus for fixing liability, once the product has been created." *Id.* at 131-32 (footnote omitted). The court will distinguish between cases where the AI or ES has been sold to an individual versus a business, rather than considering the nature of the item sold. A burglar alarm containing a microchip, programmed sensibilities and scheduling, sensors, and personal voice or palmprint code recognition may qualify as AI. A sale to an individual consumer would suffice to distinguish the holding in *New England Watch Corp.*, 11 Mass. App. Ct. at 948, 416 N.E.2d at 1010.

109. If, instead, use of the AI or ES was sold, or the results it generated were sold, then the seller could be liable for services liability or malpractice (discussed *infra* in sections IV & V of this Article).

110. Moody v. City of Galveston, 524 S.W.2d 583 (Tex. Civ. App. 1975).

111. Ransome v. Wisconsin Elec. Power Co., 87 Wis. 2d 605, 275 N.W.2d 641 (1979) (the court noted that distribution of electricity could be considered a service, but the electricity itself is a consumable good).

112. 170 Cal. App. 3d 468, 216 Cal. Rptr. 68 (1985).

113. 50 Ill. App. 3d 376, 365 N.E.2d 923 (1977).

114. *Fluor*, 170 Cal. App. 3d at 475, 216 Cal. Rptr. at 71.

115. The court in *Ransome* limited its attention to the impact of the electricity in the environment within the consumer's home. *Ransome*, 87 Wis. 2d at 622-23, 275 N.W.2d at 649.

gage Co.,[116] the court used the lender's business environment (separate computer-calculated loan statements were used to collect on loans from the same plaintiff; the company in question had offices and a loan officer in Arkansas) to conclude that the amount due on a computer-generated monthly statement could be interpreted to show intent to commit usury. (The facial interest rate was 10%, the legal maximum, but the monthly statements reflected an interest rate of 10.4712%.) However, the court in *First American National Bank v. McClure Construction Co.*,[117] distinguished the case based on the lender's business environment (the lender had neither an officer nor an agent in Arkansas and did not regularly do business in Arkansas) to conclude that the amounts due on a computer-generated monthly statement could not be interpreted to show intent to commit usury. (The facial interest rate was 10%, but the monthly statements reflected an interest rate of 10.531%.) The context of the AI or ES, and not the possible interpretations of its output, determines the potential range of liability for misapplied information. Furthermore, in *Cardozo v. True*,[118] the court found that a bookseller was not liable to a cookbook purchaser who was poisoned because the cookbook failed to indicate that one ingredient would be poisonous if left uncooked.[119] The bookseller, unlike the author of the cookbook, was not selling the information for application, i.e., the product's context was as a book rather than as a recipe to be used.

Because AI or ES is meant to exist within a particular environment or domain, the AI or ES developer cannot be held liable for the output from the AI or ES, except as it represents knowledge within the environment or domain for which it is designed.[120] To the extent that the developer is seen as a creator of information through the AI or ES, the developer cannot be held liable to all third parties for the various meanings that may be attached, absent some connection beyond the conduit of information.[121] When a domain-specific AI or ES is sold, and then a

---

116. 261 Ark. 437, 549 S.W.2d 474 (1977).

117. 265 Ark. 792, 581 S.W.2d 550 (1979).

118. 342 So. 2d 1053 (Fla. Dist. Ct. App. 1977).

119. *Id.* at 1056.

120. For example, a spreadsheet developer cannot be held liable for the number interpreted by a user as a representation of a final bid, but the developer can be held liable for its representation as the arithmetic result of the formulae input by the user. The former has a meaning outside the domain of the program (contracting bids); the latter has its meaning within the domain (mathematics) of the program. In comparison, a tax return calculating program, where the domain (tax returns for a given year) is incorporated into the product, has only one suggested interpretation.

121. Without a fiduciary duty, explicit or implicit warranty, or a context establishing expectations related to the interpretations, a contract claim is insufficient. *See* Black, Jackson & Simmons Ins. Brokerage, Inc. v. International Business Mach., Inc., 109 Ill.

buyer or user is injured by the act or information output as interpreted by the AI or ES within that domain,[122] the "product" prerequisite is met. The AI or ES, together with its actions within its domain, but independent from the meaning of the action or information external to its domain, is the "product."[123]

The product requirement has a major consequence: current medical ES do not qualify as a product as far as any patient seeking medical judgment and a course of treatment is concerned. Extant medical ES are still limited to specific narrow domains where the interpretations intended represent knowledge within that domain and not within the domain of final medical judgment towards any particular individual.[124] These programs can assist, but not supplant, the doctor's judgment concerning treatment. The patient, without direct contact with the ES in the absence of the doctor's human judgment intermediation, has no products liability cause of action, for what he seeks lies well outside the operative domain of the ES. The doctor is placed in a consultative dilemma towards the ES,[125] and the doctor or the hospital may have an indemnity claim for products liability if an injured patient's malpractice action succeeds.

App. 3d 132, 440 N.E.2d 282 (1982); Ultramares v. Touche, 255 N.Y. 170, 174 N.E. 441 (1931); Nycum, *supra* note 83, at 10-11. One might as well assert that all 'digital' representations of "4" are slanderous. Count in binary, using the fingers of either hand, and starting from either the thumb or the little finger, with the finger down to represent a "0," up to represent a "1." Raising the first digit represents "1," lowering it and raising the second represents "2," raising the first digit again as well represents "3," and lowering both and raising the third digit (inevitably the middle finger) represents "4." (It is also a lewd gesture which may provoke brawling.)

122. Since the seller can delimit the domain, he may balance the desire for a larger market (favoring expanded claims) against the desire to constrict the area for which he will be strictly liable (favoring diminished claims). If the seller, rather than specifying a domain for the AI or ES, instead states specific result guarantees, these become the source for misrepresentation actions. The requirement for domain correctness is the assurance that the product is safe. *See Negligence, supra* note 100, at 851.

123. This conceptualization of "product" may explain why on May 27, 1986, United States Patent 4,591,983 for a hierarchical knowledge system was issued to James S. Bennett and Jay S. Lark, assignee Teknowledge, Inc.

124. MYCIN's domain is consultative advice on diagnosis and therapy for infectious diseases; ONCOCIN's domain is chemotherapy; CASNET's domain is glaucoma; INTERNIST's domain is internal medicine; PIP's is kidney diseases; OWL's is digitalis therapy; HODGKINS' is diagnostic planning for Hodgkin's Disease; PUFF's is pulmonary-function interpretation; and VM is an intensive-care monitor. *See* 2 HANDBOOK, *supra* note 2, at ch. 8.

125. For an excellent discussion of this dilemma, see Willick, *Professional Malpractice and the Unauthorized Practice of Professionals: Some Legal and Ethical Aspects of the Use of Computers as Decision-Aids,* in COMPUTING POWER, *supra* note 102, at 817.

b. *Product must not be artifact.*

The majority of AI and ES built before 1986 were research projects, devices to explore the problems of artificial intelligence from a computer science perspective. The development of EMYCIN (essential MYCIN) proved that there could be a separation between an inference engine and a knowledge base. Even with the explosion of expert system shells, most programmers today design AI and ES specifically for, and sometimes with the assistance of, one particular company or customer. Knowledge engineers and support firms are springing up, and new applications are being created continually. As the field advances, the probability that mass-marketed AI or ES applications will develop reaches certainty.

As long as AI and ES remain hand-crafted, effectively, they do not qualify as products. Justice Freedman in *La Rossa v. Scientific Design Co.*[126] used this distinction to deny strict liability:

> Professional services do not ordinarily lend themselves to the doctrine of tort liability without fault because they lack the elements which gave rise to the doctrine. There is no mass production of goods or a large body of distant consumers whom it would be unfair to require to trace the article they used along the channels of trade to the original manufacturer and there to pinpoint an act of negligence remote from their knowledge and even from their ability to inquire. . . . In the present case Scientific Design's services were highly specialized and affected only the small group of employees of Witco engaged on the job. The effect of defendant's performance in supplying the pellets and supervising their installation had no element of impact on the public at large. Instead of being one of numerous public consumers of defendant's product, decedent was one of a small group of employees of Witco affected by Scientific Design's activity.[127]

In contrast, a New Jersey court, in *Schipper v. Levitt & Sons*,[128] found an element of mass production and allowed a strict liability claim against a builder of mass-produced homes with a defective heating system.

As long as the purchasers of AI or ES do not have to plumb the depths of a manufacturer-wholesaler-retailer chain, at least one driving force for the cause of action is absent. If purchasers work with system developers during installation or development, the purchaser can hardly plead lack of opportunity to discover negligence. Yet, once the element of mass-production is introduced, or the same AI or ES is sold to a large

---

126. 402 F.2d 937 (3d Cir. 1968).

127. 402 F.2d at 942.

128. *See* Schipper v. Levitt & Sons, Inc., 44 N.J. 70, 90-91, 207 A.2d 314, 325 (1965). *See also* Saloomey v. Jeppesen & Co., 707 F.2d 671, 679 (2d Cir. 1983) (designer/seller/manufacturer of mass-produced charts strictly liable).

number of customers, then the potential for a products liability cause of action increases.

2. *When Has the Defendant Sold the Product?*

The second requirement for a cause of action in products liability is that the defendant must be a seller of the product. Yet the majority of AI or ES applications built to date are individual constructions. This tailoring to specific needs and environments is likely to continue to be a major economic portion of the market for the near future. Thus, individually designed and sold AI or ES applications will not qualify as a product due to the lack of a mass market.

Another very active portion of the market today is the sale of the tools used to build AI and ES applications. Individuals who build their own AI or ES from the component parts will not be able to bring a products liability cause of action for injury from the AI or ES because the product that caused their injury was not the product sold by the manufacturer.[129] Application-building tools such as text and program editors, compilers, and expert system shells must not be confused with the computer programs that they are used to build, just as hand drills and wood planes must be distinguished from the lathes, mills, and processes that construct them. Courts must distinguish the database or initial knowledge base from the inferences attained during any given run, just as the hand drills and wood planes must be distinguished from the porch swing they are used to construct. The applications must be distinguished from the hardware, operating system, and other components in the environment in which they operate, just as the hand drills and wood planes must be distinguished from the pine, oak, screws, and workshop with which, and wherein, they operate.[130]

3. *When Does the Product Reach the Buyer Without Substantial Alteration?*

All programs transform inputs into outputs; AI and ES applications conform to this rule. This creates two distinct theoretical questions con-

---

129. *See* Greenman v. Yuba Power Prod., Inc., 59 Cal.2d 57 (1963) (involved a lathe without a proper fastening device). For a general review, see Traynor, *The Ways and Meanings of Defective Products and Strict Liability*, 32 TENN. L. REV. 3 (1965).

130. The distinction is that between object and result. The interesting problems are those that arise from flaws in the interaction between the program and the world, rather than in the production of the program. If the flaw is from something as simple (though difficult to trace) as an imperfection in the electronic copy of the program, a blank in the magnetic tape or a mislaid area in the CD-ROM disc, the theoretical concern devolves to a standard problem in products liability. Existing law can readily handle this, just as it can handle exploding coke bottles, flammable obstetrician's surgical draperies, or a flawed wheel. *See, e.g.*, MacPherson v. Buick Motor Co., 217 N.Y. 382, 111 N.E. 1050 (1916).

cerning substantial alteration: (1) when should the input supplied to the AI or ES by the buyer or user be considered a substantial alteration, and (2) where the AI or ES is designed to be able to learn and alter its own processes, when is the AI or ES's reasoning process so changed that it has undergone a substantial alteration?

a. Altered inputs.

Whenever the manufacturer identifies appropriate inputs for the AI or ES in the application's documentation, any input not meeting that description becomes suspect.[131] If the manufacturer makes an explicit statement concerning the robustness of his system over a range of possible inputs, any failure to meet this guarantee subjects him to an action for misrepresentation.[132] The user has a duty—following from the doctrine of reasonable use—to constrain his inputs to the domain in which the manufacturer specifies that the AI or ES is designed to operate. If the developer fails to inform the purchaser or user about the AI or ES implementation language capability limits, there is an implicit representation that the entire range is safe.[133] Input errors of one type or another are inevitable. Holding a developer liable for all consequences of his program, including those resulting from incorrect input, requires him to be an insurer of his product—which is beyond the limits of strict liability.[134]

At present, there is no one accepted, standard, reasonable means to

---

131. Spilling coffee into the circuitry can, on one level, be considered an input, arguably a foreseeable one, but hardly an appropriate one. The great problem here lies in the greater human power to consider analogous inputs; clever plaintiff's counsel would argue that what was actually input matched the developer's documentation because it was analogous to the examples stated; clever defendant's counsel would argue that what was actually input was not analogous because it produced a distinctly different result.

132. See Clements Auto Co. v. Service Bureau Corp., 298 F. Supp. 115, 127 (D. Minn. 1969) (defendant made representation by describing controls as "iron-clad," remaining silent as to error-proneness, and stating that system was designed to prevent the great majority of errors).

133. In Clements Auto, the court noted, "It is equally clear that [defendant] held itself out to be expert in that field, a subsidiary of IBM, the giant of the computer industry. In such a context, its failure to inform [plaintiff] of the proclivity to error of this system constitutes a representation that there was no such proclivity." Clements Auto, 298 F. Supp. at 128.

134. The possible input alphabet in Clements Auto, included numerals, some punctuation, and characters. One input field totalled 27 spaces, so its range of possible valid inputs was thus no less than 27. Up to 75 keystrokes might be required without verification. The court determined that "regardless of the skill of the operator, it would have been humanly impossible to eliminate the errors built into this system. . . . [Defendant's] system was so designed that there would always be error caused by hand keying . . . ." after concluding that it "is not the choice by [defendant of the input device] which is actionable, but [defendant's] representations about the Flexowriter operation." Clements Auto, 298 F. Supp. at 129-30.

cope with input errors, so various techniques are used: (1) reconfirmation by the user, (2) provisional, horizon-limited reasoning by the program, and (3) provisions (in an AI or ES) for error checking or error tracing. The absence of any technique may be held to be unreasonable;[135] but the existence, use, or absence of input checking may be affected by design trade-offs and economic realities at the time of creation or sale. Policy questions arise as to how strict a standard should be applied. Because the reasoning processes used by AI or ES applications do not duplicate human capacity to separate fiction from truth, the results are particularly dependant upon the input; in this one sense, machines are infinitely gullible. Even systems that maintain varied hypothetical worlds depend upon their human contacts to distinguish the uniquely real world. If the user or buyer provides false or inaccurate inputs, then the seller may raise an equitable estoppel defense.[136]

This same limitation applies to interpretation of results produced by the AI or ES. Symbolic manipulation may be designed with a particular model in mind—that is, a particular extension.[137] The possibilities for misinterpretation are limited only by human creativity and are incapable of mechanical comprehension. "Substantial alteration," therefore, in terms of interpreting results produced by the AI or ES, must be explicitly defined by the environment or implicitly delimited by the seller. If the buyer or user interprets the results beyond this environment, that would be a substantial alteration sufficient to negate this cause of action.

b. Altered reasoning.

If the buyer or user changes the way the inference engine used by the program operates, extends the knowledge base used by the program, or otherwise alters the basic nature of the program, the buyer or user would substantially alter the range of possible inputs. This type of alteration is even more possible when the AI or ES is constructed with the ability to learn and to modify itself.[138]

---

135. This is analogous to the need for a maintenance program for any automatic machinery on which human interaction depends. See Arizona Highway Dept. v. Bechtold, 105 Ariz. 125, 129, 460 P.2d 179, 183 (1969).

136. See Swiss Air Transp. Co. v. Benn, 121 Misc. 2d 129, 133-34, 467 N.Y.S.2d 341, 344-45 (N.Y. Civ. Ct. 1983) (held that the computer's limitations, which were known to the plaintiff and which enabled the forgery to be honored, justified the defense of equitable estoppel). The function of this doctrine would apply to a plaintiff who negligently provided misleading inputs; it is the nature of the input, rather than its source, that permits invocation of this defense.

137. See supra text accompanying note 4.

138. See Davis, Interactive Transfer of Expertise: Acquisition of New Inference Rules, in READINGS IN AI, supra note 18, at 410. Two features of the program TEREISIAS stand out: (1) the user, who did not have to know how the program reasoned, could provide new

Whenever the AI or ES operates with a complete and correct process, alteration of the knowledge base still can change the result drastically.[139] If the environment is not restricted to ensure monotonousness, contradictions can arise.[140] Alteration of the rules provides a wider basis for unforeseeable, unpredictable changes; if the user or purchaser changes the product, this cause of action might fail.[141] If the buyer's reason for purchasing the product is to gain the very ability to learn, which gives rise to this unforeseeable adaptability, the cause of action will not arise.[142] When manipulation of the environment is such that the product transforms itself in ways that cannot be predicted, i.e., whenever an AI or ES is capable of learning and modification which is not subject to delimitation or prediction by complete and correct means by the manufacturer, then, at the very least, a presumption of substantial alteration ought to exist.

4. *When Is the Product Defective?*

An AI or ES application without interaction problems, interpretation confusion, or input error-produced failures, but which still causes harm, does not automatically create a products liability cause of action. Determining the expectations for any product depends upon its environment—that is, upon its place in the economic world in which it is manufactured, sold, and used. There are a number of criteria which are used to evaluate each product:

> This evaluation of the product in terms of the reasonable expectations of the ordinary consumer allows the trier of the fact to take into account the intrinsic nature of the product. The purchaser of a Volkswagen cannot reasonably expect the same degree of safety as would the buyer of the much more expensive Cadillac. It must be borne in mind that we are dealing with a relative, not an absolute concept.

---

and possibly inconsistent rules; and (2) the system had the ability to ask for new knowledge and reasoning capacity where it determined it was weakest. Thus, there were two means whereby the final output of the program could be altered.

139. *Cf. supra* note 32 and accompanying text. If the only person one can reason about is Sam, and rule (3) is changed to -LUCKY(Sam), one can no longer deduce RICH(Sam). However, one will be able to determine that this is possible in a finite time.

140. For example, an employee database must allow for changes in who is or is not employed, authorized to sign documents, and the like. Default assumptions may need to be updated and temporal or situational arguments may need to be traced.

141. This defense exists even if the changes were made by a third party. *Cf.* Ward v. Hobart Mfg. Co., 450 F.2d 1176 (5th Cir. 1971) (hand guard for a meat grinder had been removed by an intervening owner; verdict for plaintiff reversed).

142. *See* Dreisonstok v. Volkswagenwerk, 489 F.2d 1066, 1071-72 (4th Cir. 1974) (plaintiff sued for injuries enhanced by lack of front protection in minibus; since this was the feature that led to purchase of that vehicle, judgment for defendants). *See also* Seattle-First Nat'l Bank v. Tabert, 86 Wash. 2d 145, 154, 542 P.2d 774, 779 (1975) (feature leading to purchase may be a question of fact).

In determining the reasonable expectations of the ordinary con-
sumer, a number of factors must be considered. The relative cost of
the product, the gravity of the potential harm from the claimed defect
and the cost and feasibility of eliminating or minimizing the risk may
be relevant in a particular case. In other instances the nature of the
product or the nature of the claimed defect make other factors relevant
to the issue.[143]

The general rule must be that when the product's design features
are the reason for the purchase, the injured party cannot raise a prod-
ucts liability cause of action.[144] Unlike algorithmic programs, AI or ES
programs do not guarantee successful completion; the best that even
complete and correct inference techniques guarantee is that if no an-
swer can be derived this fact will be revealed within a finite time. Yet,
inadequate implementation, inaccurate knowledge, or poorly designed
interaction transformations can serve as the source of defect because
"[a] defect may emerge from the mind of the designer as well as from
the hand of the workman."[145] The injury causing defect can arise from
the interaction language, the domain in which the AI or ES operates, or
the design of the inference engine. The use of an inference technique
that is incomplete, incorrect (in the mathematical sense), or (even
though theoretically complete and correct) inadequately implemented,
creates a cause of action for products liability, just as provision of a
knowledge base that proved to be inaccurate at the time of its develop-
ment would give rise to a similar products liability cause of action.
However, the time of provision would be crucial. The courts have noted
that reasonableness of design, conduct, and defects are to be measured
from the time of product design, not from the time an injury occurs.[146]

The nature of the problem may be such that failures are inevitable;
some things will never be perfect. The rule here is stated in comment k
to section 402(a) of the *Restatement (Second) of Torts*:

> There are some products which, in the present state of human knowl-
> edge, are quite incapable of being made safe for intended and ordinary
> use. These are especially common in the field of drugs. An outstand-
> ing example is the vaccine for the Pasteur treatment of rabies, which
> not uncommonly leads to very serious and damaging consequences

---

143. *Seattle-First*, 86 Wash. 2d at 154, 542 P.2d at 779.

144. If one application sacrifices completeness for flexibility or speed, and the buyer is
injured by a failure to find the correct answer, no cause of action lies for an injury caused
by a resulting malfunction. For example, if the application sacrifices incontestable inter-
pretations for a more natural language interface and the user is injured by his own misin-
terpretation, or the user sacrifices completeness for increased speed and a decreased space
limitation to make an ES operate on a personal computer rather than a mainframe, no
action would lie if these led to errors. *Cf. Dreisonstok*, 489 F.2d at 1072.

145. Greenman v. Yuba Power Prod., Inc., 59 Cal.2d 57, 134 (1963).

146. *See* Ward v. Hobart Mfg. Co., 450 F.2d 1176 (5th Cir. 1971).

when it is injected. Since the disease itself inevitably leads to a dreadful death, both the marketing and the use of the vaccine are fully justified, notwithstanding the unavoidable high degree of risk which they involve. Such a product, properly prepared, and accompanied by proper direction and warning, is not defective, nor is it *unreasonably* dangerous. The same is true of many other drugs, vaccines, and the like, many of which for this reason cannot legally be sold except to physicians, or under the prescription of a physician.[147]

There is no guarantee, or even a reasonable expectation, of any cure in medicine. Thus, there can be no cause of action for products liability for a medical ES in the near future.[148]

In *Brody v. Overlook Hospital*,[149] the court stated:

In this case the experts for both sides all agreed that in December 1966 (when the blood was transfused) there was no known scientific or medical test for determining whether blood drawn from a donor contained serum hepatitis virus. . . . In 1972, when this case was tried, the testimony indicated that the most effective test yet devised was the Australian Antigen Test which was then only about 25% effective.

. . . .

Upon the evidence present in this record we conclude that the blood transfused to decedent herein falls within the category of an "unavoidably unsafe product" and thus was not "unreasonably" dangerous

. . . .

. . . .

Based upon all of the above, we conclude that the doctrine of strict liability in tort should not be applied to the hospital or County Blood Bank in the instant case.[150]

On the other hand, if the AI or ES concerned mathematical calculations and contained a predetermined language and environment, such as a tax advisor for a given year, then the machine and human expectations may be defined relatively closely. In this case, users can easily

---

· 147. RESTATEMENT (SECOND) OF TORTS § 402A comment k (1965) (emphasis in original).

148. Some products have been defined as "unavoidably unsafe" and thus not subject to products liability. *See, e.g.*, Heirs of Fruge v. Blood Serv., 506 F.2d 841, 847 (5th Cir. 1975) (blood); Basko v. Sterling Drug, Inc., 416 F.2d 417 (2d Cir. 1969) (inadequate warning of risks of Sabin vaccine); Davis v. Wyeth Lab., Inc., 399 F.2d 121 (9th Cir. 1968) (pharmaceuticals); Ferrigno v. Eli Lilly & Co., 175 N.J. Super. 551, 420 A.2d 1305 (1980) (DES); Jones v. Minnesota Mining and Mfg. Co., 100 N.M. 268, 669 P.2d 744 (1983) (radioactive isotopes); Toole v. Richardson-Merrell, Inc., 251 Cal. App. 2d 689, 588 P.2d 326 (1967), *reh'g denied*, 251 Cal. App. 2d 689 (1967) (MER/29 not properly prepared and marketed and no adequate warning given). The grant of immunity to products other than blood depends somewhat upon warnings provided by the manufacturer to the physician, hospital, or patient.

149. 127 N.J. Super. 331, 317 A.2d 392 (1974).

150. *Id.* at 336-41, 317 A.2d at 395-97.

trace reasoning and rules and utilize computational techniques. This situation is analogous to *Blevins v. Cushman Motors*[151] where the court stated, "In our opinion, golf carts are not incapable of being made safe for their intended and ordinary use."[152]

Future jury determinations will decide what domains for ES are unavoidably unsafe. Courts have a harder problem in determining which elements of interaction with the real world remain unavoidably unsafe for AI. Ineluctably, the real world is unpredictable and hazardous, but the fact that all possible outputs of an AI or ES application may be mathematically unverifiable does not determine duty any more than does the prediction of inevitable error.[153] If there is no economic means to ensure a reasonable expectation of a safe result from an AI application, there can be no product liability cause of action. The developer would be required, however, to include the reasonable safety techniques appropriate for other designs in the field at the time of development and sale.[154] At this intermediate level, the jury must determine whether the defendant's judgment in failing to provide a safeguard was reasonable—assuming, of course, that the plaintiff established that it was more likely than not that the safeguard would have prevented the injury. Finally, when the developer can provide inexpensive intermediate verification of the input or result that would have precluded the injury, and the cost of providing such activity in the AI or ES is minimal, he has a duty to provide such safeguards,[155] unless the purchaser makes it clear that he does not desire them.

A developer must provide an adequate interface in order to have a successful ES.[156] The interface may be such that the user has a nearly natural language communication capacity, or it may use special graphical representations to convey spatial or other domain-specific knowledge. There can be particular problems with design and

---

151. 551 S.W.2d 602 (Mo. 1977).

152. *Id.* at 608.

153. "Foreseeability alone . . . creates no duty. If such were the case, a manufacturer of hammers, foreseeing injured fingers and thumbs, would be liable for every such injury. This duty is established as a matter of social policy—as a means to an end." Wilczek, *Products Liability-Manufacturer Has No Duty to Design an Automobile Frame Which Will Protect Occupants in a Collision*, 42 NOTRE DAME L. REV. 111, 115 (1966) (citation omitted). *Accord* Dreisonstok v. Volkswagenwerk, 489 F.2d 1066 (4th Cir. 1974).

154. *See* Maietta v. International Harvester Co., 496 A.2d 286 (Me. 1985) (plaintiff alleged that one of the possible causes of several accidents was a failure in circuitry of computerized anti-skid braking system; products liability cause of action was upheld).

155. *Cf.* Helling v. Carey, 83 Wash. 2d 514, 519, 519 P.2d 981, 983 (1974). The court held that: a duty to test exists when testing can avoid a grave and devastating result; the test is simple and relatively inexpensive; there is no judgment factor involved; there is no doubt that the test detects the defect; and giving the test is harmless. *Id.*

156. *See* BUILDING ES, *supra* note 2, at 225-57.

implementation of a specific interface. As stated previously, the AI or ES engages in transformations within the interaction language. This language serves the need to compromise between the uncertainties and imperfections of the real world and the mathematical verities that can be manipulated by the hardware. As such, it must preserve the essential isomorphisms of the domain, but may do so at the expense of connotative richness.

The developer must use transformations that are reasonably isomorphic (i.e., not reasonably vulnerable to misinterpretation), such that the essential features are preserved. Failure to do so led to liability in *Aetna Casualty and Surety Co. v. Jeppesen & Co.*[157] Here the defendant presented vertical and horizontal graphical views with two different scales for an airport approach chart. The court stated:

> The plan view is regarded as a superior method of presenting course and course changes; the profile view as a superior method of presenting altitude and altitude changes. Each chart thus conveys information in two ways: by words and numbers, and by graphics. . . . The "defect" in the chart consists of the fact that the graphic depiction of the profile, which covers a distance of three miles from the airport, appears to be drawn to the same scale as the graphic depiction of the plan, which covers a distance of 15 miles. In fact, although the views are the same size, the scale of the plan is five times that of the profile.
>
> . . . .
>
> While the information conveyed in words and figures on the Las Vegas approach chart was completely correct, the purpose of the chart was to translate this information in an instantly understandable graphic representation. This was what gave the chart its usefulness— this is what the chart contributed to the mere data amassed and promulgated by the FAA. It was reliance on this graphic portrayal that Jeppesen invited.
>
> [T]he Las Vegas chart "radically departed" from the usual presentation of graphics in the other Jeppesen charts; . . . the conflict between the information conveyed by words and numbers and the information conveyed by graphics rendered the chart unreasonably dangerous and a defective product.[158]

Should the buyer request a particular implementation or interac-

---

157. Aetna Casualty & Sur. Co. v. Jeppesen & Co., 642 F.2d 339, 342 (9th Cir. 1981).

158. *Id.* at 342. This case does not illustrate a difference over interpretations of the output, but a problem with how the output is displayed or stated, i.e., the interaction language. If internal predicates are translated to natural language, then the concern is with proper choice of translations and a reasonable balance between flexibility and precision. If the information is displayed in a graphic format (e.g., pie chart, bar chart), then the dimensions and assignments must be accurate, and the scale must be distinct to avoid the problems seen in *Jeppesen*. Since many AI and ES applications will involve complex graphical presentations, this case should serve as a warning to carefully screen and evaluate these presentations.

tion language, the developer is under a duty to warn about potential problems that may arise.[159] The developer can provide a warning in a number of ways, such as a glossary of the implementation language, explicit and detailed documentation of the assumptions inherent in the AI or ES, descriptions of the domain and environment of operation, or implementation of an explanation facility to allow the user to spot flaws in the justification for the result reached by the program.[160] Whenever the developer states such a limitation, an injury produced by an erroneous, mistaken, or improper interpretation should not be actionable.

This constriction of strict liability would not apply where the manufacturer has the opportunity and the ability to predict or constrain the set of possible inputs, i.e., where the task-language is rigidly delimited. There are many situations in which input or output can be typified. For example: a spreadsheet expects to manipulate numbers, not letters; a question's answer must be "yes" or "no," not "maybe"; and the possible range for some input devices are pre-set. Failure that occurs when the answer is out of range can hardly be ascribed to the computer; it is the responsibility of those who determined the possible limits.[161] When the acceptable answer or input can be rigidly delimited, errors resulting from the failure to check for accuracy may be an appropriate basis for an award in products liability.

5. *When is the AI's or ES's Defect the Source of the Injury?*

One line of cases prohibits imposition of strict liability when no personal injury is involved, regardless of the nature of the transaction.[162] Kansas specifically denied recovery for economic loss caused by

---

159. *See* Clements Auto Co. v. Service Bureau Corp., 298 F. Supp. 115, 128 (D. Minn. 1969), *aff'd in part, rev'd in part*, 444 F.2d 162 (8th Cir. 1971) (defendant had superior knowledge about the probable flaws in the implementation desired by plaintiffs).

160. *See* Davis, *Interactive Transfer of Expertise: Acquisition of New Inference Rules*, in READINGS IN AI, *supra* note 18, at 413.

161. At present, there is a debate over the long-term failure by a weather-reporting satellite to report the existence of a hole in the ozone layer over the South Pole. The satellite's computer program was instructed to classify as absurd the readings it was getting. After ground observers reported the figures, scientists had to re-examine all of the satellite's transcriptions. Perhaps the failure here was that the satellite's program was too limited (it lacked a comparison-over-time feature, or the ability to report repeated impossible readings) or that the scientists' and programmers' definition of impossibility in the real world was erroneous.

162. *See, e.g.*, Industrial Risk Insurers v. Creole Prod. Serv., 568 F. Supp. 1323 (D. Alaska 1983) (after the pipeline company's pump station exploded causing only property damage, the court granted summary judgment to the firm that provided the design for the pump station); Seeley v. White Motor Co., 63 Cal. 2d. 9, 403 P.2d 145 (1965) (denying recovery for commercial injury but granting it for property damage). *But see* J'Aire Corp. v. Gregory, 24 Cal. 3d 799, 598 P.2d 60 (1979) (granting economic losses in a negligence action).

not in the performance within the domain but with the purchaser's ex-
pectation for commercial gain, the injury is not caused by the AI or ES.
Even when the AI or ES meet each of the elements above, the equitable
principles that underlie this cause of action must be reviewed to deter-
mine whether or not an award is justified; examination of these princi-
ples as they apply to products liability for AI or ES is therefore in
order.

C. UNDERLYING PRINCIPLES AS APPLIED TO AI OR ES

One analyst believes that products liability may serve as an ano-
dyne for nonprogrammers: "The need to understand programming tech-
nicalities creates inequalities in bargaining power between the computer
professional and the client."[165] However, the principle of overcoming
inequality in bargaining power only permits products liability to be in-
voked when the court feels contractual barriers were imposed based on
imbalanced knowledge, market power, or unconscionable practices. It
does not indicate whether, after the cause of action is invoked, a judg-
ment ought to be granted. The balance between a developer's and
user's abilities to provide possible and appropriate environments and in-
puts, respectively, circumscribes the expectations society is willing to
support by the imposition or nonimposition of strict liability. From
what principles are these limits derived in particular cases?

There are at least four basic principles[166] that support a judgment
based on products liability: (1) the stream of commerce principle; (2)
the control of risks principle; (3) the risk cost-spreading principle; and
(4) the deep pocket principle. Of these four, some have better reasoned
support and more vocal proponents than others which have a less ar-
ticulated but wider felt basis. To do justice in accordance with each
principle's inner warrant requires attention to both the logic and emo-
tion underlying the varied jury and judicial decisions that articulate
that principle.

1. *Stream of Commerce or Commercial Product?*

The stream of commerce principle[167] assumes that the AI or ES de-
veloper voluntarily made his product available to the general public for

---

165. *Disclosure, supra* note 100, at 558.

166. *See* Calabresi, *supra* note 96; Fletcher, *supra* note 96; Traynor, *supra* note 129.
For a discussion of strict liability with regard to computer programs in general, see
Gemignani, *supra* note 15, at 195-203. For a discussion of the applicability and possible
defenses of products liability, see Nycum, *supra* note 83, at 15-20; Freed, *supra* note 100;
Schneider, *supra* note 99, at 548-50 (relevant only to defective home computer programs).

167. *See Negligence, supra* note 100, at 850; Note, *Continuing the Common Law Re-
sponse to the New Industrial State: The Extension of Enterprise Liability to Consumer
Services,* 22 UCLA L. REV. 401 (1974) (authored by Jon Chait).

his economic gain. It is not applicable to all commercial products; rather, it is limited to those products generally considered to be consumer goods.[168] This principle will not support products liability for an AI or ES that is not made available to the general public or that has a nature such that only a special subset of the general public ever will be interested economically in the application.[169] Neither is the stream of commerce principle generally applicable when there is doubt whether a sale or a product is involved—although a rough analogue is used for service liability. Since this principle considers transaction costs,[170] a plaintiff, seeking to bolster his claim by reference to those costs, requires solid market-related economic evidence to serve as the basis for his claim to this ethos of justice.

There are two rationales supporting this principle. The first rationale is that potential gain to the general public of having a new, and presumably better, product must be balanced against its unforeseen hazards, and the individual injured members must be recompensed. Thus, the chief burden of the indirect cost is more justly assessed against those that directly profit from the new product. The second rationale is that the manufacturer's invitation to use his product contains an implicit assurance that the product is safe. Therefore, consumers who place their faith in the superior knowledge and judgment of the manufacturer ought to be protected.

The balancing test tips the scales against the use of the stream of commerce principle to invoke products liability for two reasons. First, imposing strict liability may distort true market costs (which include resolution of just claims for recompense for flaws in the marketed product) and constrict the market so that a new advance, not yet matured, will be crushed by imposition of strict liability's transaction costs. This reason (used in the nineteenth century to protect infant steel and railroad industries) looks to the societal increase of wealth gained from new products and industries. Second, when bargaining parties have roughly equivalent capacity and knowledge, the ethos of fairness that supports balancing gain and risks is served best within the bargaining process. Thus, the courts should decline to impose an unexpected trans-

---

168. New England Watch Corp. v. Honeywell, Inc., 11 Mass. App. Ct. 948, 948, 416 N.E.2d 1010, 1011 (1981) ("no one can seriously contend that this subject matter [a burglar alarm system] constitutes 'consumer goods'"; and, because that prerequisite was missing, the cause of action for products liability was dismissed).

169. For example, General Electric Company created CATS, an ES used to diagnose and repair diesel-electric locomotives; Teknowledge Company created SECOFOR, an oil well drill-bit-sticking advisor; and Westinghouse Company created an ES for nuclear fuel enhancement. *See* B. BUCHANAN, *supra* note 2.

170. *Cf.* Calabresi, *supra* note 96, at 1094.

action cost.[171] The second reason against imposing products liability pursuant to a stream of commerce principle is particularly appropriate for special interest and limited domain AI and ES. If a domain expert designs the AI or ES for use as a sorcerer's assistant or a performance-enhancing tool, or if the customer is involved in the specification and building process, a products liability cause of action would be subsumed by commercial warranties (explicitly or implicitly stated or omitted).[172]

One author strongly argues that strict liability should not be assessed against software, including AI and ES software.

> It appears that the phenomena of software programs and computer output might not be exactly what non-lawyers have been calling them. Hence, the applicability of products-liability rules on the basis of analogy cannot be determined without an understanding of those phenomena. With that understanding, it appears that the analogy most immediately identified is not apt. Software programs seem to be processes and computer output seems to be streams of electrons. The question then arises whether the products-liability approach ought to be expanded into these areas, as a matter of social policy. Until greater perfection can be achieved in software program design, if it ever can,

---

171. *See, e.g.*, Southwest Forest Indus., Inc. v. Westinghouse Elec. Corp., 422 F.2d 1010, 1013 (9th Cir. 1970) (dispute over a computer failure between a power company and the generation plant contractor); Salt River Project Agric. Improvement and Power Dist. v. Westinghouse Elec. Corp., 143 Ariz. 437, 694 P.2d 267 (1983); Kaiser Steel Corp. v. Westinghouse Elec. Corp., 55 Cal. App. 3d 737, 748, 127 Cal. Rptr. 838, 845 (1976); Scandinavian Airlines Sys. v. United Aircraft Corp., 601 F.2d 425, 428 (1979).

In addition to bargaining power, the court in *Salt River Project* emphasized the opportunity each party would have to spread losses:

> SRP and Westinghouse are each large commercial enterprises who were dealing from positions of relatively equal bargaining strength. The purchase of the LMC was conducted in a commercial-industrial context in which each side had an opportunity to deal with the other in negotiating the sale and its terms. Further, both are business entities who sell a product or perform a service which is ultimately paid for by Westinghouse's or SRP's customers. As a result, "[w]hether the loss is thrust initially, upon the manufacturer [Westinghouse] or customer [SRP], it is ultimately passed along as a cost of doing business included in the price of the products of one or the [services provided by the] other and thus [is] spread over a broad commercial stream."

Salt River Project, 143 Ariz. at 442-43, 694 P.2d 272-73 (citing *Kaiser Steel Corp.*, 55 Cal. App. 3d at 748, 127 Cal. Rptr. at 845)).

172. *See* La Rossa v. Scientific Design Co., 402 F.2d 937 (3d Cir. 1968). Some artificial intelligence companies have manufactured an expert system builder and created knowledge engineering teams, and a large portion of their business comes from assisting their clients to create, maintain, or expand an expert system (Teknowledge and SI, Intellicorp and KEE). The inference engine and the database manipulation routines are one product, but the actual knowledge and the final ES are another. Failure to include a knowledge building aspect or explanation function might be seen as an overall flaw, but this may have been traded for speedier performance or serve as a market-distinguishing feature. *Cf.* Davis, *Interactive Transfer of Expertise: Acquisition of New Inference Rules*, in READINGS IN AI, *supra* note 18, at 410.

there might be good reason to avoid saddling that new industry with intolerable liability responsibilities. Since computer output vulnerabilities stem in large part from software program deficiencies, it well might be appropriate to spare the data processing industry from that broad exposure as well. Both industries, still infants, are extremely important to contemporary society and should not be stunted by such burdens.[173]

2. *Control of Risks or Control of Environment?*

The control of risks principle is stated as follows:

The supplier . . . is in a better position to anticipate and control the risks of harm. As a result, the supplier is in a better position to determine if the product is safe enough for use by the public. Because of this presumed judgment capacity of the supplier, public policy requires that he be held liable for any injury that results when the product is not safe.[174]

This principle assumes that the supplier is able to anticipate and control the risks of harm; it inadequately addresses the problem of harm arising from interaction with the wrong external environment. The driving force behind this is again a concern over how to allocate efficiently the costs of changes in our society. This was articulated by California Chief Justice Traynor in *Escola v. Coca Cola Bottling Co.*: "[P]ublic policy demands that responsibility be fixed wherever it will most effectively reduce the hazards to life and health inherent in defective products that reach the market. It is evident that the manufacturer can anticipate some hazards and guard against the recurrence of others, as the public cannot."[175] Traynor's opinion noted that only some hazards can be anticipated and guarded against by the manufacturer. Pragmatic considerations create distinct limitations on when this principle can be used to favor a cause of action for products liability. Two potential defenses, whereby the manufacturer establishes that the hazard was not one he could reasonably anticipate or guard against, are product misuse and assumption of risk.[176]

Each product in our society is assumed to carry with it a common sense understanding of proper use. Hammers are not used for anaesthesia, and razor blades are not used for floor tiling. Floppy disks are not frisbees—if used as the latter, a purchaser cannot seek recovery for his lost programs or data. The product misuse exception recognizes that a manufacturer should be held liable for only a subset of the possible

---

173. Freed, *supra* note 100, at 285.

174. *Negligence, supra* note 100, at 850 (footnote omitted).

175. Escola v. Coca Cola Bottling Co., 24 Cal. 2d 453, 462, 150 P.2d 436, 440-41 (1944).

176. For a discussion of these with regard to software, see Gemignani, *supra* note 15, at 200-04.

environments and uses of his product.[177] Assumption of risk is recognition that the user must accept some responsibility for limiting (or not limiting) his risk-taking behavior. AI and ES may reach the consumer, but there will always be problems with real world or human interactions over which their designers have no control.[178]

The control of risks principle presumes that the supplier can anticipate and control risks using his or her knowledge of society and basic common sense. Therefore, when a risk is a feature of the environment in which the product is used, and reasonable safety features are incorporated, this principle will not support an award. The presumption of the developer's superior knowledge in the general case becomes subordinate to the user's superior knowledge in a particular instance; the question becomes one of where the balance of knowledge lies. Also, if the risk is an unexpected real world or domain complexity that requires human common sense, the user cannot expect the AI or ES to evaluate that risk. Common sense is precisely what cannot yet be assumed for any AI or ES. Situations outside the predefined environment or domain, or involving unpredictable, anomalous, or extraordinary truths, produce nonsensical results that betray a lack of common sense. At present, and for the foreseeable future, the user ineluctably has the burden of providing common sense and appropriate inputs.[179]

The complexity of the hardware and software environment in which the AI or ES operates calls into question the presumption that the supplier can anticipate and control the risks. If the developer states a limitation upon possible computer environments,[180] inputs, or interpretations for the knowledge used by the AI or ES, to delimit the range

---

177. For example, the manufacturer is not liable when his lawnmower is used as a hedge-trimmer, even though this is a possible use—as is use of a lawnmower as an anchor or a decorative chair. The range of possibilities is limited only by the human imagination.

178. Examples include entering the wrong figures from incorrectly obtained specifications, or opening a new cell in a spreadsheet program without altering the formulae in the other cells to accept the additional information. *See generally* International Paper Co. v. Farrar, 102 N.M. 739, 700 P.2d 642 (1985).

179. *See, e.g.*, Ford Motor Credit Co. v. Hitchcock, 116 Ga. App. 563, 158 S.E.2d 468 (1967) (defendants held liable when, rather than depending on their own knowledge, they relied on what the computer told them); Ford Motor Credit Co. v. Swarens, 447 S.W.2d 53, 57 (Ky. App. Ct. 1969) ("Trust in the infallibility of a computer is hardly a defense, when the opportunity to avoid the error is as apparent and repeated as was here presented."); Neal v. United States, 402 F. Supp. 678, 680 (D.N.J. 1975) (a possible tax refund error claim arose because "what was at work was the GIGO Rule of Computers (Garbage In, Garbage Out)").

180. Though it may well be that the AI or ES will work without difficulty in many look-alike environments, the manufacturer cannot be held to have stated this. For example, the AI or ES designer may have neither knowledge of, nor control over, the compiler used by the purchaser; this fact can affect the quality of the application's performance. *See supra* text accompanying note 10.

of knowledge, the developer should be able to constrain his corresponding liability.[181] This limitation on liability not only encourages accurate delimitation (and thus serves the policy of limiting risks to society generally) but also limits consumer incentives to remain ignorant or heedless.

3. Risk Cost-Spreading or Preventive Effort Allocation?

The risk cost-spreading principle is motivated entirely by economic ethos with no regard for intent. Its basic presumption is that errors and injuries are inevitable. Thus, whenever the manufacturer is in a better position than the purchaser to spread the cost of such injuries (that is, whenever the manufacturer sells to enough purchasers that the manufacturer can adjust the price to cover the direct and transaction costs) this principle supports imposition of strict liability.[182]

Analysts may accurately predict one thing about any computer application: somehow, somewhere, the application will fail.[183] One analyst remarked:

> Specific programs can be tested to reveal "bugs" that must be corrected, however, no amount of testing can guarantee that all the bugs in the program have been found; even after extensive testing, the program, which in a major software project can be extremely complicated and expensive, may still fail miserably. . . . There are, of course, various safeguards built into programs and computers which attempt to catch errors and avoid major damage, but there always remains some chance of catastrophic failure. It is not a question of whether there is

---

181. This line of reasoning is similar to that used to determine the adequacy of a warning, rather than the sufficiency of an explicit warranty. *Cf.* Kammer v. Lambs-Grays Harbor Co., 55 Or. App. 557, 639 P.2d 649 (1982), *petition denied*, 293 Or. 190, 648 P.2d 852 (1982) (products liability upheld, not because the computer operating a paper-roll moving machine was faulty, but because the manufacturer failed to warn against the possible danger of built-up air pressure activating the roll kicker even after the computer was turned off).

182. "In effect, the cost of the injury is made a cost of the product. In this way the cost of the injury is spread over all the buyers of the product." *Negligence, supra* note 100, at 850-51.

183. In IBM v. Catamore Enter., Inc., 548 F.2d 1065, 1068 n.5 (1st Cir. 1976), *cert. denied*, 431 U.S. 960 (1977), the defendant's expert witness declared that the process of creating software required "tests to identify and eliminate the inevitable human error encountered in writing instructions." The witness stated that a payroll program generally requires 20,000 explicit instructions (which might take a programmer a year to code) and might contain an erroneous instruction in every 200 to 250 instructions.

In Chatlos Sys., Inc. v. National Cash Register Corp., 479 F. Supp. 738, 748 (D.N.J. 1979), *aff'd*, 635 F.2d 1081 (3d Cir. 1980), *cert. denied*, 457 U.S. 1112 (1982), plaintiff alleged that defendant fraudulently promised success, but the court noted that: "[p]laintiff's own expert testified that 40% of all computer installations fail. It is not unreasonable to view all such installations as somewhat of an experiment."

some risk, but of whether the level of risk is acceptable.[184]

A second analyst noted:

> To remove all errors from a computer program, therefore, is a difficult if not impossible task. Original program errors may go undetected for months or even years—for example, if the error becomes apparent only under a unique set of circumstances. Moreover, modifications of a computer program may introduce new errors, or cause previously present but unnoticed errors to become apparent. The difficulty in detecting errors may be compounded when the modifications are performed by programmers and analysts other than those who initially created the program. . . . Many of these functions could still be performed manually, but are performed faster, more economically, or more accurately when done by computer.[185]

The plaintiff testified that computers can manage tasks that people cannot in *Emerson v. Empire Fire & Marine Insurance Company*.[186] Thus, the basic presumption underlying the cost-spreading principle seems assured—the focal question being whether the manufacturer ought to or can appropriately adjust the product's cost. Yet, to permit a cause of action to depend on the marketplace's current status, rather than upon the relative responsibility and ability of the manufacturer and customers to prevent an injury, ignores a desirable function of the law: to encourage and support preventive efforts. The marketplace and industry standards cannot sufficiently promote any cause of action because manufacturers might choose to market products with inferior safety features or use substandard practices. The court in *The T.J. Hooper* ruled against this reasoning.[187]

The question actually turns more on the nature of the product. Inasmuch as inevitable limitations prove to be the source of the injury, neither the manufacturer nor the purchaser of the program can prevent the injury; it is a function of the real world and would be inevitable even if the computer were replaced by a team of individuals. Because the purchaser is as free to buy the product as the manufacturer is to sell it, there is little reason to insist on transferring liability from the purchaser to the seller, especially given the attendant societal cost of enforcement.

The situation is more problematic when the purchaser's freedom is constrained more tightly. Marshall Willick suggests that professionals may be required to purchase ES applications to avoid successful malpractice charges.[188] With such a guaranteed market, societal balance

---

184. Gemignani, *supra* note 15, at 185-87 (footnotes omitted).

185. *Defective Programs*, *supra* note 108, at 126.

186. 393 So. 2d 691 (La. 1981).

187. The T.J. Hooper, 60 F.2d 737 (2d Cir. 1932), *cert. denied*, 287 U.S. 662 (1932).

188. Willick, *Professional Malpractice and the Unauthorized Practice of Professionals:*

might require developers to accept liability for their products, enforcing a continued market incentive to successfully control product quality or to improve products. Under such circumstances, this policy could support imposition of a products liability cause of action.

4. *Deep Pocket or Budding Industry?*

Practitioners, researchers, students, and observers of the fields of artificial intelligence, computer science, and expert systems agree on two propositions: these fields represent new areas of human, societal, and economic endeavor; and the amount of talk, puffery, hype, and consequent expectations exceeds the objectively measured results. The underlying theoretical and mathematical basis is hardly more than a century old; the means for physical implementation are relatively recent; the first commercial ventures are hardly more than a handful of years old.[189] Yet, already there are annual international scientific conferences and dozens of competing journals (both research and commercial). In addition, hundreds of companies, or departments within companies, investigate, produce, and market products that may be called AI or ES, if not by their salespeople, by a lay observer. These two observations indicate that the industry is evolving very quickly—possibly faster than the legal system can adapt to the new challenges. Certainly legal analysts are deeply divided as to the way the legal system should address this rapidly evolving field.[190]

Money—the deep pocket—is the motivating factor for many, if not most, product liability suits. Protection of a new venture is, and has been, a motivation for denying extension of a cause of action beyond its current limits. These two motivations come into direct conflict in this field. Resolution of this conflict is not, and cannot be, based on any stated reason, law, or logic, because the conflict extends from a contradiction inherent in our society. Resolution will be based on the determinations of juries, serving as a source of knowledge of what justice is in our society. This is the genius of the system, however ineffable and impalpable, and it should not be slighted. In establishing the causal link between product and injury, the conduct and behavior of plaintiff and defendant should be examined with an eye toward the relative ability of each to apply human judgment, common sense, and reasoning to

Some Legal and Ethical Aspects of the Use of Computers as Decision-Aids, in COMPUTING POWER, *supra* note 102, at 817.

189. Charles Babbage suggested his "Analytical Engine" in the early 19th century. Frege created his logic in the late 19th century. Shockley invented the transistor and McCarthy invented LISP at the end of the 1950s. Both Teknowledge and Intellicorp are corporations of the 1980s.

190. *Compare* Schneider, *supra* note 99, at 554-55 (strongly favoring imposition of strict liability) *with* Freed, *supra* note 100, at 50 (strongly opposing imposition of strict liability).

prevent all possible harms. The focus should not be on the particular harm itself since both parties in litigation have the benefit of 20/20 hindsight for the flaw revealed. If the question devolves into one of human capacity, then the plaintiff, as well as the defendant, must share responsibility and be expected to cope with the unexpected.[191] If the problem is one inherently beyond human capacity, then the question is whether the advance was reasonably protected against human folly.

D. CONCLUSION, EXAMPLES, AND TRANSITION

The conclusion to be drawn from the foregoing analysis is that products liability may or may not be an applicable cause of action. Like all difficult questions, its resolution depends upon the facts of each case. Thus, it will be pleaded more often than it will be proven, with the defendant bearing the burden of demurring or otherwise demolishing an inapplicable plea. One thing is certain, it presents a difficult balance of responsibility for results between random plaintiffs and wide-ranging interactions, and, consequently, encourages more conscious and conscientious behavior on the part of both. Some insight may be gained by considering the following hypotheticals, which are intended to serve as starting points for informed and reasoned discussion.

1. *The Automated Chemical Factory*

The scale of the automated chemical factory is such that it would almost certainly be a unique, individually designed project. Therefore, products liability would not be an applicable cause of action because the mass market and, most likely, imbalance of knowledge prerequisites would be absent. There may be some portions of the AI implementation that could be actionable—specifically, the vision-and-movement interaction used by maintenance machinery. An injury arising from a collision between a human and a robot platform could serve as the basis for the allegation. The problem here is in determining what the vision-and-movement programmer could reasonably design as the default assumption. A default requirement for collision avoidance runs dual risks of noneconomic false alerts and of interrupting emergency shutdowns, where the prevention of a potentially explosive situation or other mass disaster becomes more important than avoiding a temporary injury. If the vision-and-movement AI subportion were designed for potentially hazardous environments (e.g., factories or nuclear power plants), the lack of an alterable default might be considered a defect; the question would then turn on the economic reasonableness of the developer's incorporating this into the particular design. However, if the vision-and-

---

191. Anyone who has ever programmed a computer or worked with one knows a sad truth: they do exactly what they are told to do.

movement AI subportion were designed for general utility, then this emergency problem would be beyond its context or presumed environment, and therefore, the subportion would not qualify as a defect at all.

2. *The Home Environmental Monitor*

The mass market requirement is met by the home environmental monitor, as is the imbalance of knowledge requirement regarding system operation. However, definition of any medical threshold is an area where the imbalance no longer holds. Because the significance of the threshold is medically determined, it is beyond the context of the monitor, and not fit for a products liability cause of action, if a mishap arises from a continued condition that does not actually violate the go/no-go threshold condition (e.g., diabetic coma). Direct human override provisions for programmed lighting, heating/cooling, or washing systems schedules should neither disrupt the schedule nor be prevented. This action ought to be reasonably foreseeable because human schedules are flexible, probabilistic forecasts. This type of input alteration is foreseeable even though the particular alterations are not. A burglary involving a break-in which did not follow reasonably foreseeable positions (e.g., cutting through otherwise solid surfaces) and did not trigger an alert, should not create a cause of action since prevention of this type of circumvention is not within the realm of the AI developer and is not reasonably expected within the realm of home environments. Similarly, any inappropriate reactions to a deliberate invocation of some code word by the owner would not be actionable (e.g., false code for burglary leading to a false arrest or a false fire alarm raised in a family dispute) because the monitor is incapable of discerning human intention. Moreover, a failure to respond definitely would lead to a claim against the monitor's developer.

3. *The Pharmaceutical Interaction Advisor*

Presuming correct use of, and a proper underlying environment for, the ES, sufficient grounds may exist for a products liability cause of action in the pharmaceutical interaction advisor context. The mass market and imbalance of knowledge prerequisites are met; constriction of the market (fear of malpractice drives pharmacists to purchase the ES) and completeness of the domain make an error circumscribed. The possibility of harm arising from mistakes in the input reasonably would require existence of procedures to prevent that harm. If input errors were the sole cause of the injury, then there would be no defense based on unforeseeable misuse.

Different facts would lead to different results, however. If the source of the patient's harm was the interaction of the prescription drug

with an illegally used drug, then causation becomes suspect because this situation is beyond the ES's stated competency. Either the pharmacist knew of the illegal drug, and then unreasonably followed the advice of the ES knowing it to be potentially inaccurate, or the pharmacist was reasonably capable of penetrating the patient's deception concerning illegal drug use. Rationales for this balancing test include: (1) the foreseeability across society that accounting for some use of illegal drugs is certain (which would favor imposition of liability), and (2) the desirability of imposing liability as an incentive to uncover deception where there is a possibility of preventing harm (which would disfavor imposition of liability). Given such a balance, market conditions of ES, beyond the one in question, may be a deciding factor. Courts should not impose liability if it will result in a greater loss to society due to constriction of progress.

On the other hand, a court should not grant an award to a patient harmed by the use of an ES with an outdated knowledge base. The safety features of the ES, as well as the knowledge of the developer, are measured as of the time of creation, not at the date of harm. Market forces, including the pharmacists' desire for up-to-date information, would spur creation of an ES which could improved, and taught, while fear of unexpected and unpredictable future liability would slow improvements that otherwise might prevent injuries.

4. The Financial Advisor

Again, the prerequisite of a mass market product is met, but, in the context of the financial advisor, there is less certainty that an imbalance of knowledge exists. If the cause of the injury lies in interpretation of a common term (e.g., profit) or from failure of the ES to include a common sense economic assumption (e.g., to budget for a vacation or provide for unexpected health expenditures) which a human advisor would have included, the balance is far from certain. A court may potentially limit the realm of possible interpretations or assumptions: a human advisor may have communicated the same advice, then judged from external clues the need to include additional warnings—feats beyond the ES's capacity.

If a developer creates incorrect production rules giving rise to tax liability or penalty without a warning that the matter was beyond the ES's capacity, the plaintiff would have a cause of action. Because the developer would have superior knowledge—the ability to prevent the harm—imposing the award would encourage future corrections. This injury, though economic in nature, is hardly commercial; it goes beyond the damage represented by the underlying arithmetic error. Unless the

failure was so grossly negligent as to amount to malice (in the legal sense), punitive damages would not be appropriate.

If the source of the injury lies in a mistaken input (e.g., a wrong earnings figure), a plaintiff could not state a cause of action unless the mistake could be categorized as impossible within the domain. Any rational dollar sum, rounded to two digits, would be possible below limits that likely would call out a production rule to seek human advice—for example, individual earnings of several billion or more. Identifying the sum as impossible, given the other figures provided by the user, would require knowledge of the human social environment beyond the capacity of the ES. Failure to perform such reality checking would not give rise to a cause of action. The user would be aware of problems that might arise from inaccurate inputs and have the capacity to recognize and make corrections. Depending on the market, internal trade-offs between speed and certainty, and the hazards involved in a mistake (i.e., the most important rule preconditions), the existence or nonexistence of an input verification interaction feature might be considered actionable, as might a provision of an ambiguous result representation. However, the failure that could give rise to liability here would not lie in the inability to identify probable inputs, but rather, in the failure to provide an interaction facility equal to the internal verities.

What about the individual who is adversely affected by the AI or ES owned by another, but is only injured because of the information, data, knowledge, or activities he himself provided? When the injured individual's activity is the source of harm, there may be a second, more applicable cause of action for service liability.

IV. SERVICE LIABILITY: MOST APPLICABLE BUT ILL-DEFINED

A. Prerequisites for Service Liability

1. *Services Covered by This Cause of Action*

Service liability is a relatively new theory evolving from products liability. This cause of action is growing out of cases where the line between *product*, *service*, and *sale* is fuzzy—precisely those cases where AI and ES are most likely to be involved. Presently, it is easier to define the types of transactions that are not included in the cause of action than those types of transactions that are included. First, service liability does not include failure to perform according to a level of competence implicitly warranted by the behavior of those in the profession who regularly engage in such transactions. These transactions are covered by the malpractice cause of action and are discussed below. Second, service liability does not include failure to perform according to particular explicit warranties. Also, services, unlike products, are not

subject to the contractual and other requirements of the Uniform Commercial Code (U.C.C.),[192] and thus are not subject to actions under this area of law. This limits, somewhat, the extensibility of implied warranties and other claims based on contract. Finally, service liability does not include any transaction where the plaintiff purchased the defendant's hourly labor. The purchaser must buy more than mere performance of a task; there must be an inherent quality in the activity or service which requires the purchaser to accept a higher charge for the seller's labor and the seller to assume responsibility for his performance. Without this feature, no economic support exists to justify imposition of the additional cost of strict liability. These twin prerequisites were stated in *Gagne v. Bertran*[193] where the court noted:

> [T]here is nothing in the evidence to indicate that defendant assumed responsibility for the accuracy of his statements. . . . [T]he amount of his fee and the fact that he was paid by the hour also indicate that he was selling service and not insurance. . . . The evidence in the present case does not justify the imposition of the strict liability of a warranty.[194]

Thus, the service agreements that are covered are those not covered by extant warranties (either explicit or societally implied). The covered service agreements include: those where the purchaser is buying more than just labor and those where the purchaser can expect judgment, and acceptance of responsibility for service quality, from the seller.

2. *The Nature of the Service*

In *Magrine v. Krasnica*,[195] after the hypodermic needle used by the defendant dentist broke off in the plaintiff's jaw, the plaintiff sought to charge the dentist with strict liability. The court denied the action, stating:

> Novelty, of itself, does not foreclose consideration of plaintiff's contentions in this field of developing tort law. . . . Neither does it justify a headlong leap to impose strict liability unless, based on proper policy considerations and reason, such liability should be found. Plaintiff concedes that there is no precedent—anywhere—holding a dentist, or any other 'user' of an article, strictly liable for injuries caused by a latent defect therein.
>
>   . . . .
>
> A dentist or a physician offers, and is paid for, his professional

---

192. Computer Servicenters, Inc. v. Beacon Mfg. Co., 328 F. Supp. 653 (D.S.C. 1970), *aff'd*, 443 F.2d 906 (4th Cir. 1971) (contract for performance of data processing services: analysis, collection, storage, and reporting of certain data was not a sale of goods and not within the scope of the commercial code).

193. 43 Cal. 2d 481, 275 P.2d 15 (1954).

194. *Id.* at 487, 275 P.2d at 19-20.

195. 94 N.J. Super. 228, 227 A.2d 539 (1967).

services and skill. That is the essence of the relationship between him and his patient.

. . . .

Defendant dentist is not in the business of supplying needles.[196]

Magrine cannot be read to prohibit a service liability action against the developer of an AI or ES program. First, in *Magrine*, the possible malpractice cause of action protected against potential negligence. But, if a service liability cause of action was brought against an AI or ES developer or manufacturer, there would be no action for malpractice available to block the service liability pleading. Second, the *Magrine* court declined to find a cause of action for service liability because the defendant neither distributed the hyperdermic needles, nor had control over their latent defects. However, the AI or ES developer is in the business of providing the AI or ES and manifests some control over it as well.[197]

However, if mistaken input or use in a hardware and software environment distinct from that specified by the developer, was the defect that caused the injury, the defect would be considered a latent one over which the developer had no control, and then *Magrine* would apply. Otherwise, the court noted:

> We must consider, also, the consequences if we were to adopt the rules of strict liability here. The same liability, in principle, should then apply to any user of a tool, other equipment or any article which, through no fault of the user, breaks due to a latent defect and injures another. It would apply to any physician, artisan or mechanic and to any user of a defective article—even to a driver of a defective automobile. In our view, no policy consideration positing strict liability justifies application

---

196. *Id.* at 235-36, 227 A.2d at 540-44 (footnote omitted). The court's reasoning, stating premises, inference rules, and particular assignments of value, echoes one format used by AI and ES programs:

> Plaintiff's argument moves from the major premise that "strict liability" is not confined to "sales," through the minor premise that the basic policy considerations of the doctrine apply to the use of a needle by a dentist, and concludes that he should be held liable though free from negligence. Since the major premise is established . . ., it therefore remains for us to analyze the policy considerations. . . .

Id. at 235, 227 A.2d at 541. However, the human ability to analyze policy considerations and projections of future impact, based on understanding of human society and judicial common sense, is far beyond what the rule-determined control structures and variable assignment techniques used by AI and ES today can readily simulate.

197. This is a previously hidden assumption: that the injured party is suing the creator, developer, producer, or manufacturing team that brought the AI or ES into being, rather than any programmer, expert, or team that maintains the AI or ES. Both *Hillas v. Westinghouse Elec. Corp.*, 120 N.J. Super. 105, 293 A.2d 419 (1972), and *Johnson v. William C. Ellis & Sons Iron Works, Inc.*, 604 F.2d 950 (5th Cir. 1979) would exempt those maintaining the AI or ES from products liability and thus, potentially, from services liability.

of the doctrine in such cases. No more should it here. . . .[198]

A year later, a Texas court considered a similar problem. In *Barbee v. Rogers*,[199] the defendant offered prescription, fitting, instruction, sale, and care of contact lenses at multiple outlets for standard prices. The plaintiff alleged improper fitting and instructions and pleaded negligence, breach of implied warranty, and products liability—bracketing the concepts from which service liability cases arise. Reversing in favor of defendant, the court took note of the nature of the defendants, the domain, and the transaction.

> [T]he activities of Respondents fall between those ordinarily associated with the practice of a profession and those characteristic of a merchandising concern. A professional relationship is present in the facts that contact lenses are prescribed and fitted by a licensed optometrist after examination of the eyes and in the exercise of his judgment. A merchandising relationship is suggested by the multiple offices of Respondents throughout the State; by their advertising and sales techniques designed to promote the sale of contact lenses at a predetermined and advertised price; and by their standardization of procedures and methods.[200]

Barbee paid attention to the skill and judgment necessarily used by the defendants—precisely the elements that are captured in an ES—and the fact that the individual circumstances of each purchaser's eyes affected the design and fitting of the contact lens.

> [T]he fact remains that the contact lenses sold to Petitioner were designed in the light of his particular physical requirements and to meet his particular needs. Presumably, and insofar as this record shows, they were not in existence when Petitioner sought the services of Respondents. .They were not a finished product offered to the general public in regular channels of trade. The considerations supporting the rule of strict liability are not present.[201]

These elements are present in each consultation or situation involving the ES or AI, but there are commonalities, predictable ranges, and general rules from which the particular decision or action is derived. In 1968, the type of clinic run by the defendant in *Barbee* was unusual; today, optometrists, doctors, lawyers, dentists, financial advisors, and other human servicers are found marketing their skills in various clinics or as adjuncts to department stores. Market pressure will probably induce some standardization and produce pre-set responses according to the type of customer, thus smoothing out individual circumstances.

While the market between individual and mass-market services is

---

198. *Magrine*, 94 N.J. Super at 241-42, 227 A.2d at 547.
199. 425 S.W.2d 342 (Sup. Ct. Tex. 1968).
200. *Id.* at 345-46.
201. *Id.* at 346.

growing more blurred, a distinction still can be made. If the service cannot be standardized, whether from too many variables or from the need to account for a constantly and unpredictably changing environment, then a plaintiff should be unable to bring a cause of action for service liability. Equally, no ES or AI program should be produced other than as a research project. Alternatively, whenever a service is identical across a class of inputs or situations, the potential for strict liability exists[202] and an ES or AI program is potentially feasible. If a developer markets an AI or ES program without a documented statement of the identity classes on which it operates, then a cause of action may be justified even though the user applies it to a situation outside the boundaries assumed by the developer or manufacturer.

The key factor in *Barbee* was not the defendant's role as a supplier of the defective product, but the defendant's required professional status.[203] Within the year, a New Jersey court allowed a cause of action for service liability in *Newmark v. Gimbel's Inc.*[204] The court held that a cause of action for strict liability could be brought against a beauty parlor operator for injury to a customer's hair allegedly resulting from use of a permanent wave solution. The court noted that if the solution, rather than its application, were sold, a cause of action for products liability would have existed. The cost of the solution was a part—but only a part—of the whole transaction; the court characterized it as "hybrid partaking of incidents of a sale and a service."[205] This hybrid was sturdy enough to support an implied warranty of fitness both for the product and for the skill and knowledge of the beauty parlor operator, whom the customer expected to avoid harmful products. The *Newmark* court specifically distinguished the facts in *Newmark* from those in *Magrine* on the basis that the beauty parlor operator in *Newmark* applied the product to the plaintiff and treated her desires rather than her needs.[206] The underlying presumption of individual judgment distinguished the dentist's actions in *Magrine* from the beauty parlor operator's actions in *Newmark*.

---

202. *Cf.* Johnson v. Sears, Roebuck & Co., 355 F. Supp. 1065 (E.D. Wis. 1973) (allowed a cause of action for strict services liability against the hospital for mechanical and administrative services provided by hospitals); Schwartz v. United States, 230 F. Supp. 536 (E.D. Pa. 1964) (castigated the Veterans Administration for negligence in administrative failure to trace identity of patients exposed to a hazardous treatment and to obtain and employ plaintiff's past medical records, available in the same building, saying that the Government cannot interpose the defense that its right hand did not know what its left hand was doing).

203. Texas law provided that contact lenses could be made available only through a licensed practitioner.

204. *See* Newmark v. Gimbel's, Inc., 54 N.J. 585, 258 A.2d 697 (1969).

205. *Id.* at 593, 258 A.2d at 713.

206. *Id.* at 595, 258 A.2d at 714.

> [The dentist's] performance is not mechanical or routine because each patient requires individual study and formulation of an informed judgment as to the physical or mental disability or condition presented, and the course of treatment needed. . . . Such men are not producers or sellers of property in any reasonably acceptable sense of the term. In a primary sense they furnish services in the form of an opinion of the patient's condition based upon their experienced analysis of the objective and subjective complaints, and in the form of recommended and, at times, personally administered medicines and treatment.[207]

A line of reasoning that distinguishes professional services from other services might explain the courts' choices to date. Dentists and optometrists are professionals by virtue of social history and state legislation; beauticians are not.[208] This rough delimitation has been followed in California,[209] Wisconsin[210] and New York, where one court stated bluntly:

> Absent a guarantee of specific results, those engaged in the professions are . . . required only to use due care in the performance of the professional services rendered. They may be held in malpractice for the negligent performance of their professional services but in this state no cause of action is known to the law against an architect for a breach of implied warranty. . . . Warranty being the basis for the doctrine of strict products liability, it follows that the cause of action will exist only where there would have been warranty implied, but as discussed above, there is no implied warranty in connection with professional services. There is no cause of action in strict products liability for damages allegedly resulting from the negligent performance of architectural services.[211]

The reasons for this distinction, however deeply rooted in our socie-

207. *Id.* at 596-97, 258 A.2d at 715.

208. Although beauticians must be licensed in California, CAL. BUS. & PROF. CODE § 7320 (West Supp. 1989), special licensing is needed to perform a permanent wave. *Id.* § 7354.1.

209. *See* Pancoast v. Russell, 148 Cal. App. 2d 909, 307 P.2d 719 (1957) (architect); Roberts v. Karr, 178 Cal. App. 2d 535, 3 Cal. Rptr. 98 (1960) (surveyor); Bonadiman-McCain, Inc. v. Snow, 183 Cal. App. 2d 58, 6 Cal. Rptr. 52 (1960) (engineer); Lindner v. Barlow, Davis & Wood, 210 Cal. App. 2d 660, 27 Cal. Rptr. 101 (1962) (accountant); Gautier v. General Tel. Co., 234 Cal. App. 2d 302, 44 Cal. Rptr. 404 (1965) (commercial services); Allied Properties v. John A. Blume & Assoc., 25 Cal. App. 3d 848, 102 Cal. Rptr. 259 (1972) (marine engineers).

210. *Compare* Schuster v. St. Vincent Hosp., 45 Wis. 2d 135, 172 N.W.2d 421 (1969) (court declined to impose a higher standard of care than ordinary care against a hospital with regard to custodial functions) *with* Johnson v. Sears, Roebuck & Co., 355 F. Supp. 1065 (E.D. Wis. 1973) (hospital held strictly liable for its alleged failings in administrative or non-medical activities).

211. Queensbury Union Free School Dist. v. Jim Walter Corp., 91 Misc. 2d 804, 398 N.Y.S.2d 832, 833-34 (1977) (architects) (footnotes omitted). The same result was reached in *Hall v. New York*, 106 Misc. 2d 860, 435 N.Y.S.2d 663 (1981) (although for a different reason: governmental immunity).

tal strata, are poorly articulated and directly challenged by potentials inherent in ES and AI applications. It is precisely when there are predictable, duplicable, and generalizable classes of behaviors or reasoning that these applications become possible and advances come so swiftly, making what was expert yesterday standard today. Furthermore, licensing is used to limit access to a profitable source of income as well as to ensure a certain level of competence. The success of many individuals in convincing state legislatures to create a new profession should not be the sound and reasoned basis for the imposition (or lack thereof) of strict liability for an AI or ES program's activities.

If a human's advice or action causes an injury, as long as that expert was practicing a profession, no strict liability cause of action would accrue. This is true, according to this distinction, whether the reasoning was mechanical or would be identical across a broad class of circumstances. On the other hand, a non-professional service provider who acts with the same degree of care would be subject to strict service liability. The degree of care, or the need to consider the distinguishable elements, does not govern this difference in liability. Instead, it is the fortuitous existence of a recognized socially warranted profession. The ES or AI application could perform at a level either equal or superior to the human with regard to the problem, and yet be limited in expertise and therefore unable to qualify for licensing.[212] Should it be judged by the professional or nonprofessional standard? Rather than depend on the existence of societal institutions—which are not capable of reacting swiftly enough to adapt to the technological advances producing the ES and AI—we are better served by looking to the underlying quality of performance and potential for improvement.

Determining whether a cause of action for service liability does, or should, exist for ES or AI applications requires deeper consideration of: (1) what constitutes professional as opposed to standard services; (2) the policy considerations that support or oppose imposition of strict liability rather than negligence; (3) the extent to which AI or ES response reflects the basic nature of the problem; and (4) the success or failure of the implementation used by the creator of the AI or ES to copy or model the domain and the reasoning process used by humans: expert, professional, or merely common sense.

The evolution of our society is beginning to produce service operations which emulate the transition of production operations, during the

---

212. For example, none of the medical or other expert systems mentioned in the introduction could be considered capable of meeting the licensing requirements for a doctor in any state. Yet, performance in the domain for which they are designed is as good as experts in the field and superior to general practitioners. It is simply their lack of breadth that renders them incapable of being licensed. What will happen when future broad-base expert programs are devised and implemented?

first half of this century, toward organized mass production. Strict liability for services would be appropriate when the following criteria are met: (1) the service is marketed to a large number of individuals; (2) the service is identical across distinguishable classes or individuals, rather than requiring and reflecting specific circumstances to be performed for each purchaser; (3) the service is of such a definable and delimitable nature that, given the circumstances of the purchaser, human experts reasonably would not produce different services; and (4) the service is a voluntary interaction whose principle motivation is economic. An AI or ES that meets these four criteria might give rise to a cause of action against the developer or manufacturer, just as it may give rise to a cause of action against a human service provider. Whether such a cause of action should be allowed requires examination of the policies underlying imposition of strict service liability.

B. POLICY CONSIDERATIONS FOR SERVICE LIABILITY

1. *The Extensiveness of Those Affected: Special Application or Mass-Effect?*

All services formerly lacked two crucial requirements of a products liability cause of action: mass production and distribution. While today's economy is justly described as a "services economy" with the availability of services of all types greatly expanded, a distinction remains between services that require human judgment and consideration of the unique context of the process (such as negotiation between adversaries)[213] and services that can be standardized for classes of consumers, situations, or inputs. The element of mass production led a New Jersey court to allow a strict liability claim in *Schipper v. Levitt & Sons, Inc.*,[214] against a builder of mass-produced homes with a defective heating system, while the Third Circuit used this same distinction to deny strict liability in *La Rossa v. Scientific Design Co.*[215]

The AI or ES applications considered here are not those which are individually crafted, research tools, or single-user applications. Rather, they are applications where a number of individuals either can purchase the identical application or are affected by the use of one application, i.e., those that have definite effect upon the consumer populace rather

---

213. Analysts have discussed a program simulating (on a conceptual level) a negotiator; but there is no suggestion of any capacity to replace the human individual's ability to discern and project the proper (i.e., persuasive and effective) emotive adjuncts to the conceptual argument such as voice tones, inflections, timing, body language, or silence. The additional ability to induce the background, concealed, or subconscious motivations of the adversary and thus work with greater contextual comprehension is another uniquely human skill.

214. 44 N.J. 70, 207 A.2d 314 (1965).

215. 402 F.2d 937 (3d Cir. 1968); *see supra* note 172 and accompanying text.

than a minuscule submarket. Just as the lack of mass production exempts items from products liability actions, so should lack of mass-servicing exempt an AI or ES application from a service liability action.

2. *The Nature of the Underlying Domain: Predictable or Inherently Uncertain?*

Certain services are of such a nature that they are historically recognized as uncertain. The classic examples of this are services relating to medical treatment or legal representation. But courts also note that computer installations and subsequent operations are far from certain successes.[216] Improved tools, even improved reasoning tools, only lead to an increased examination of the environment and the proper place of the tool, rather than of the nature of the service as a source of liability.

The distinction between service and products liability diminishes as the mechanical nature of the former increases. Rather than categorize services as "professional" or "nonprofessional," the underlying domain should be examined to determine whether the task is predictable or inherently uncertain. AI or ES applications depend upon a domain being predictable, even if this requires an artificial restriction, i.e., created by the work of man. When the domain's model is essentially isomorphic to the domain, then there is potential for automating the proper responses; when no model can hope ever to capture or predict the domain adequately, then there is less potential for automating the proper responses.[217] Although a plaintiff cannot be expected to provide complete and correct proof that a model is perfect, i.e., that the task of predicting behavior within that domain can be mechanical, a plaintiff can offer a basic societal test that should suffice: If the average human worker within the domain would be considered a "technician" rather than a "professional," then the domain is not inherently uncertain for the purposes of imposing service liability for an AI or ES application.[218]

Medicine is the subject of many assaults upon its professional status—perhaps because there is certainty of personal injury which inspires visions of rewards to be gained from susceptible juries.[219] Many

---

216. *See supra* text accompanying note 183.

217. However, for service liability concerns, something less than mathematical Platonian perfection may suffice.

218. One scholar argued: "Strict liability is especially appropriate for mechanical tasks, which are typically so easily done correctly." Greenfield, *Consumer Protection in Service Transactions—Implied Warranties and Strict Liability in Tort*, 1974 UTAH L. REV. 661 (1974).

219. There is an equal certainty of injury in any lawsuit that is not settled—one side loses. However, the lesser aura of scientific certainty, a reluctance to assail members of the same profession (that formerly extended to fellow professionals such as doctors), and the lack of competition have limited the actions against attorneys by their fellows. The

courts have asserted that the domain of medical practice is inherently uncertain.[220] Yet there is a growing distinction between the uncertainties which may affect that domain and those which may affect the quality of care received by the patient. This distinction is sufficient to impose strict liability when certain prerequisite features exist.

In *Johnson v. Sears, Roebuck & Co.*,[221] the court refused to disallow all strict liability for professional medical services provided by the defendants (doctors and hospital) when plaintiff alleged defective but non-negligent services. The court reasoned as follows:

> Initially, hospitals provide at least two types of services. The first consists of professional medical services and the second is made up of those mechanical and administrative services which support the first.
>
> It is argued, since strict tort liability should not apply to professional medical services by doctors . . ., that it follows that strict liability should not apply to mechanical and administrative services by hospitals. I do not think this follows. Medical sciences are not exact. A patient cannot consider a doctor's treatment to be defective simply because it does not cure his ailment. All that a doctor can be expected to provide is adequate treatment commensurate with the state of medical science. In other words, doctors do not contract with patients to provide cures but rather to provide treatment in a non-negligent manner. To hold medical professionals strictly liable under these circumstances would not promote any social benefit. In fact, if that standard were applied to doctors, it might make then reluctant to assume responsibility for the treatment of patients, particularly when such treatment involves a developing area of medicine, which would work a serious social disservice.
>
> I do not, however, feel that the mechanical and administrative services provided by hospitals should necessarily be exempt from strict liability.[222]

---

latter two circumstances are changing; the former may or may not change depending on advances in computerizing the legal profession and the law.

220. *See, e.g.*, Barbee v. Rogers, 425 S.W.2d 342, 345 (Sup. Ct. Tex. 1968) ("The acts of prescription and fitting are described as an art with many variables and call for an exercise of judgment by the practitioner").

In *Newmark v. Gimbel's, Inc.*, 54 N.J. 585, 596-97, 258 A.2d 697, 703 (1969), the court noted that "[n]either medicine nor dentistry is an exact science; there is no implied warranty of cure or relief. There is no representation of infallibility and such professional men should not be held to such a degree of perfection. There is no guaranty that the diagnosis is correct."

In *Hoven v. Kelble*, 79 Wis.2d 444, 469, 256 N.W.2d 379, 391 (1977), the court stated that "[t]here are differences between the rendition of medical services and transactions in goods (or perhaps other types of services as well). . . . [P]rofessional services tend often to be experimental in nature, dependant on factors beyond the control of the professional, and devoid of certainty or assurance of results."

221. Johnson v. Sears, Roebuck & Co., 355 F. Supp. 1065 (E.D. Wis. 1973).

222. *Id.* at 1066-67.

The court, in arriving at this conclusion, considered the seriousness of the consequences of defective services, the nearly total inability of the plaintiff to recognize or control the defective service, and the public interest in minimizing the defects of those which, because of the underlying domain, were inherently uncertain.

Professional services which are distinguished by uncertainty, that is, those where identical inputs may produce different decisions from human experts in the field, or where the result depends on qualities which are not reducible to limited formulae, are not subject to strict liability. There is an underlying uncertainty in the domain that the average practitioner—even the very best practitioner—cannot overcome. Allowing strict liability would hold human practitioners to a standard of performance that is impossible to attain. However possible a perfect solution might be in each case, there is no possibility for every case to have a perfect solution. Therefore, if the AI or ES is performing in such an area (as would be the case with medical diagnostic ES, legal case analysis ES, or human factor-predicting financial analysis ES), the field's imperfection permeates the performance of the ES so that imposition of service liability for the wrong result would be inapposite.

There are areas where the domain is, or can be, so well-defined and the rules of decision are, or can be, so well-refined that, given identical inputs, decisions of the human experts would not differ.[223] By proving that the AI's or ES's domain of practice was of such a nature, a plaintiff should be able to bring an action based on strict service liability for the ES's failure to meet the human performance, as long as other general policy and particular situation considerations, discussed below, are met.

3. *The Nature of the Interpretation Between Representation and Real Events*

If an injury occurs because of an imperfect implementation of the ES or AI, rather than because the domain is incapable of providing a more certain answer, then a distinction, akin to that drawn by the *Johnson* court, becomes possible. Still, a second hurdle remains: Is the failure caused by the selection of the representation or in the implementation of the reasoning process? The performance of programmers, knowledge engineers, and supporting experts, requires ingenuity and creativity;[224] through ingenuity and creativity some elements

---

223. Examples include analysis of gases within human blood samples, identification of fingerprints, and financial analysis based on defined, stated, objective factors. The areas of perfectible expertise generally are areas of technical expertise requiring more knowledge, skill, and education than is possessed by the average individual, but are not beyond their capabilities or requiring judgment based on domain knowledge (as opposed to assessment of the correctness of the procedures used to obtain the data).

224. There are many unsolved problems in the area of knowledge representation that

will be perfectible: (1) explicit delimitation of underlying assumptions, ranges of accurate behavior, and application environment requirements; (2) identification of supporting data on which representation and rules are based (to human authors or external studies); and (3) specification of trade-offs made between performance and comprehensibility (e.g., speed vs. breadth). Transcription into an appropriate representation will not be mechanical, at least in the foreseeable future, but ensuring correct and predictable behavior by the inference engine might be.

If an application is potentially incomplete, or capable of drawing inconsistent conclusions because the inference is mathematically incapable of perfection, then failure to inform the purchaser would be an implicit warrant that the incompleteness would not affect the results—a guarantee that the remoteness of the possibility would not happen. If the remote event does occur, however, then a cause of action for strict liability could be argued pursuant to *Johnson*.

In contrast, when the error's source is inherent ambiguity or limitations forced upon the developers to represent the domain with terms limited to computer precision, i.e., where the transcription to a computer-recognizable representation must generalize over connotative distinctions or where the knowledge is either uncertain or inconsistent, then a cause of action for strict liability could not arise. A human actor could not possibly further circumscribe the potential for injury by any better or more complete representation.

Similar considerations of the representation problem are required when the cause of the injury arises from the interaction between the human user and the application, rather than from the advice or action given by the program. This problem focuses on the interpretation placed upon the advice or the reaction to the computer's actions. Here, a broad spectrum of questions arise as to the ability, or lack thereof, of the plaintiff (or the computer) to recognize or control the defective service. Just as a hospital should not be held liable for an outpatient's failure to take prescribed medicine, an AI application should not be held liable for a human's failure to ignore a recognizably hazardous situation, and an ES application should not be held liable for a human user's unorthodox interpretation of its advice. If courts impose strict liability without considering these questions, they will place the entire burden of anticipating human reactions upon the program developers. This burden can be justified only if the ability to recognize or control the situation is beyond the capacity of the average human being or the predefined average user. Even then, if the representation is such that it

---

concern the acuity and accuracy of mapping between the real world and the model internal to the application. Presently, there is not even the suggestion of a mechanical means to perform this task independent of human judgment.

requires special skills or training to comprehend it in the manner antic-
ipated and provided for by the developer, and such requirements are
stated, the nonspecialist user who applies a different interpretation has
no cause of action for strict liability. Even technicians are not required
to explicate their results in common English; technical tasks typically
require a refined and defined subset of the language for precise compre-
hension. Laymen cannot bring suit for failure to explain or resolve
problems in terms they can understand, or lawyers would find them-
selves universally accused.

4. *The Nature of the Demand: Voluntarily Sought or Compelled Condition?*

Courts will deny a strict service liability claim when the service is
such that it is considered compulsory, that is, when the service is one
that would be sought by any individual in the purchaser's circum-
stances. The policy reason for distinguishing between services meeting
individual choices and those necessary for all individuals in like circum-
stances reflects the interplay between economics and equality. Courts
must consider this interplay because imposition of strict liability affects
all of society and thus impacts far more than any individual plaintiff's
and defendant's past transaction.

If the court determines that the service is one any person in plain-
tiff's position might seek but that the plaintiff had the choice of
whether to obtain it, then market balancing may potentially justify the
increased liability of those potential defendants who provide the service.
On the other hand, if the service is one which must be encouraged to
serve the general welfare, a different policy is suggested. The court in
Newark v. Gimbel's[225] stated:

> The beautician is engaged in a commercial enterprise; the dentist and
> doctor in a profession. The former caters publicly not to a need but to
> a form of aesthetic convenience or luxury, involving the rendition of
> non-professional services and the application of products for which a
> charge is made. The dentist or doctor does not and cannot advertise for
> patients; the demand for his services stems from a felt necessity of the
> patient. . . . [Doctors' and dentists'] unique status and the rendition of
> these sui generis services bear such a necessary and intimate relation-
> ship to public health and welfare that their obligation ought to be
> grounded and expressed in a duty to exercise reasonable competence
> and care toward their patients. In our judgment, the nature of the
> services, the utility of and the need for them, involving as they do, the
> health and even survival of many people, are so important to the gen-
> eral welfare as to outweigh in the polity scale any need for the imposi-

---

225. 54 N.J. 585, 258 A.2d 697 (1969).

tion on dentists and doctors of the rules of strict liability in tort.[226]
Similarly in *Hoven v. Kelble*,[227] the court considered it important that
"[m]edical services are an absolute necessity to society, and they must
be readily available to the people."[228] A permanent wave is a consumer
option which some choose and some do not; correction of visual acuity,
while probably desired by all, reflects an element of personal desire
when contact lenses are chosen over glasses; but medical treatment for
life-threatening situations is sought by all persons. When the nature of
the demand is such that any individual in the plaintiff's situation would
demand the same service, there is less possibility that economic factors
will support imposition of strict liability as a means to provide better
quality services.

    This policy does not come directly into play where AI and ES appli-
cations are concerned. In the past, courts balanced the certainty of a
service's market against society's desire to avoid limiting the number of
practitioners, often resulting in a regulated industry. Even in cases
where there is compulsory or universal demand, market forces still af-
fect selection of the service provider. The transformation of hospitals
from eleemosynary institutions to for-profit corporations proves this
point. AI or ES applications offer the service provider the opportunity
to improve the quality or decrease the cost of services, thereby creating
a competitive advantage. The hospital or individual medical practi-
tioner is the more likely purchaser for any AI or ES, rather than the
patient. Although the element of necessity is removed, concern about
the necessity to provide the best treatment may reinstate some of this
compulsory demand.[229] Since existence of AI or ES programs may af-
fect market selection of a service provider, the market is arguably the
best means to drive provision of accurate and well-designed AI or ES
programs. Thus, imposition of strict liability is not barred, even when
the service is compulsory. Yet the same concern remains valid. If im-
position of strict liability upon the AI or ES manufacturer penalizes the
use of their programs—programs the courts have begun to recognize as
more efficient than duplication of human expertise by duplication of

---

226. *Id.* at 597, 258 A.2d at 702-03.

227. 79 Wis.2d 444, 256 N.W.2d 379 (1977).

228. *Id.* at 469, 256 N.W.2d at 391. *Cf.* Johnson v. Sears, Roebuck & Co., 355 F. Supp.
1065, 1067 (E.D. Wis. 1973) ("[If strict liability] were applied to doctors, it might make
them reluctant to assume responsibility for the treatment of patients, particularly when
such treatment involves a developing area of medicine, which would work a serious social
disservice.").

229. Marshall S. Willick, Esq. unveils the dilemma practitioners may face of having to
use expert systems or be charged with malpractice via an extension of the rule announced
in The T.J. Hooper, 60 F.2d 737 (2d Cir. 1932). *See* Willick, *Professional Malpractice and
the Unauthorized Practice of Professionals: Some Legal and Ethical Aspects of the Use of
Computers as Decision-Aids*, in COMPUTING POWER, *supra* note 102.

human effort, and therefore, worthy of support—then potential improvements may never be made.[230]

Direct comparison to human experts' or actors' exposure to, or immunity from, strict liability is inadequate. If a human actor providing the same service as the AI or ES is not subject to liability, it is arguable that the developer should not be subject either. Unlike the human expert, who is one person, an AI or ES program can be duplicated and thus work in multiple locations and provide far greater revenues. Presently, there is a bottleneck limiting production of AI or ES applications and a limited comprehension of what is considered a good or effective design. With the plethora of new tools coming on-line, this may be a temporary phenomenon. The rationale limiting strict liability for doctors, dentists, or other experts meeting a universal demand in order to avoid driving away practitioners, and thus limiting service availability, may fade into oblivion with the coming of this service revolution. With these changes, any ban on those developing the applications composing this revolution will fade.

In summary, policy considerations which may affect imposition of, or exemption from, strict service liability for an AI or ES application include:

(1) whether the scope of the service provided by the application is limited to a minuscule fraction of those members of society who might be interested, i.e., whether there is an element of mass application;

(2) whether the underlying domain is inherently uncertain, preventing any human actor as well as any AI or ES application from circumscribing the risk of harm, or whether performing the service can be a mechanical situational response and whether that need is communicated to the user or purchaser;

(3) whether the cause of injury arose from the interpretation or action from the injured human or from the AI or ES application, and whether there is any potential for a perfectible isomorphic representation between the computer model and real events; and

(4) whether the demand for the service is so universal that the need to encourage the service outweighs the benefit of using strict liability to weed out inefficient or imperfect applications.

---

230. Courts recognize this principle when considering computer programs that assist professionals, even though they are far from AI or ES quality. *Cf.* Wehr v. Burroughs Corp., 619 F.2d 276, 284 (3d Cir. 1980) (awarded costs for use of computer-aided legal research systems because such use is "certainly reasonable, if not essential, in contemporary legal practice"); United Nuclear Corp. v. Cannon, 564 F. Supp. 581, 592 (D.R.I. 1983) (specifically praised the use of such applications); Golden Eagle Dist. Corp. v. Burroughs Corp., 103 F.R.D. 124, 129 (N.D. Cal. 1984), *rev'd on other grounds*, 801 F.2d 1531 (9th Cir. 1986) (chastised counsel for unreasonable failure to cite adverse authority despite access to LEXIS, a computer-aided legal research system).

Additional policy considerations that affect individual cases (regardless of AI or ES application involvement) were considered in *Magrine v. Krasnica*[231]:

> Warranties may be imposed or annexed to a transaction by law, because one party to the transaction is in a better position than another (1) to know the antecedents that affect . . . the quality of the thing . . . dealt with; (2) to control those antecedents; (3) and to distribute losses which occur because the thing has a dangerous quality; (4) when that danger is not ordinarily to be expected; (5) so that other parties will be likely to assume its absence and therefore refrain from taking self-protective care.[232]

Pragmatic considerations in particular cases that may affect imposition of service liability, as these reflect the above concerns, are examined below.

C. PRAGMATIC CONSIDERATIONS FOR PARTICULAR CASES

1. *The Class of Defect Must Be Correctable or Preventable*

The chief rationale for imposing strict liability, as opposed to any other form of liability, is to provide an incentive to change and to prevent other injuries to other individuals. Advancing this rationale is a prerequisite to imposing service liability in a particular case. Analysts have argued that service liability is inappropriate for individuals because the service is not correctable. This reasoning forbids service liability if the problem caused by the AI or ES application is the product of a unique and nonreproducible environmental combination.[233] When the source of an injury can be traced to a reproducible and correctable software flaw ("bug"), however, this argument would not bar imposition of service liability. Brannigan and Dayhoff summarize this point:

> Since services are performed at a particular point in time, defective services, even the services that involve products, are not generally considered to have correctable defects. While the effect of a defectively performed service can sometimes be countered by performing additional services, the original service no longer exists, and thus there is no chance for adjustment. Products, such as machines, on the other hand, can be adjusted, repaired, or altered, even long after production. The process of debugging a program is very similar to the process of adjusting a machine. In addition, the program can be handed over to others besides the systems analyst or programmer for debugging. In

---

231. 94 N.J. Super 228, 227 A.2d 539 (1967).

232. *Id.* at 232, 227 A.2d at 541 (quoting 2 HARPER & JAMES, LAW OF TORTS § 28.19, at 1576 (1956)).

233. *Cf.* Freed, *supra* note 100, at 274-75. As stated in the introduction, there are a large number of software failures that are nonreproducible bugs. In fact, the apocryphal source of the phrase arose from a failure of the ENIAC computer caused by a moth crawling in the central processor and inadvertently short-circuiting the computer.

this functional comparison of characteristics, a program is very similar
to a product.[234]

Courts should impose service liability only if more than the particular piece of code, choice of representation, or method of implementation is correctable. Certain types of services simply are not of a nature where the error which gave rise to an injury, when corrected, ensures greater future security. This includes a service whose success is affected primarily by its external environment, such as medical advice, environmental monitoring and maintenance, information-processing advising programs supporting professional (that is, judgmental) services, or in-factory movement patterns. If the environmental interaction is highly variable, the risk that correction of the error will give rise to one or a host of further errors is too great. Services that are an incidental or peripheral part of another transaction would also be exempted[235] because correction to meet incidental requirements could seriously impede capability in the main area of interest.

Imposing service liability in a particular case, therefore, requires some showing that the actor—here, the AI or ES—could act differently and, given the impetus of paying for damages, will act differently in the future. If the machine cannot change its program, which makes the error not merely reproducible but also inevitable, then it cannot be held liable after the first example. Further problems would arise from human failure to account for machine incompetency. This premise has the necessary corollary that the developers or marketers for the AI or ES must make every effort to educate their customers concerning known problems where human judgment is required, at the risk of bearing strict liability for failure to communicate that need for judgment.

2. *The Nature of the Service: Mechanical Situational Response or Creative Action?*

The law already recognizes that the area of human design is exempt from service liability because creativity, awareness of the particulars of the environment, and use of common sense within the context of future interaction, are all required. When the process of designing is at issue, rather than the resulting product, courts recognize that the service is not subject to the liability that the product might be. In *Stuart v. Crestview Mutual Water Co.*,[236] the court denied strict liability against

---

234. *Defective Programs, supra* note 108, at 132.

235. *See* Black, Jackson & Simmons Ins. Brokerage, Inc. v. IBM, 109 Ill. App. 3d 132, 440 N.E.2d 282 (1982) (court granted summary judgment against a plea for negligent misrepresentation on the grounds that the defendant principally sold hardware to the plaintiff and was not in the business of supplying information).

236. *See* Stuart v. Crestview Mut. Water Co., 34 Cal. App. 3d 802, 110 Cal. Rptr. 543

the engineering company which designed, engineered, and built water systems that subsequently proved to have inadequate pressure to support fire-fighting needs.

Unlike standard programs, AI or ES applications can be designed to learn, and can produce incompletely understood and unforeseen results. If the application is sold with an open-ended design (i.e., one that uses imperfect or incomplete inference mechanisms, or that has modifiable databases, use assumptions, or some form of default reasoning, or that is capable of learning and modifying its own behavior accordingly), then the user is purchasing what effectively adds up to a creative, not a mechanical, response to new situations. Unlike a three-piece suit, scissors, or even a word processor, the AI is capable of producing unforeseen results. Indeed, if the system is open-ended, even the designer cannot predict its later capacities. Since the strength of the AI or ES is provision of a more general purpose tool for its user than the standard computer program, the purchaser does not receive a standard service; instead, the purchaser buys a creative or adaptive service. As such, no cause of action for service liability lies. It should be noted that the focus is not on whether an AI or ES is involved, but rather on the nature of the particular AI or ES involved—the very quality of detail that ought to explain distinct outcomes based on general legal principles.

3. *Intervention of Human Judgment Breaks the Causal Connection and Prevents Imposition of Strict Liability*

In *Swett v. Gribaldo, Jones & Associates*,[237] the plaintiff homeowner sued the defendant soils engineer for damages to his home caused by soil instability. Although the defendant had performed the preliminary grading, he withheld final approval pending final grading and construction. The plaintiff did not consult with defendant further or provide it with any opportunity to review or approve the foundations, the final grading plans, or the results, which altered the slope and nature of the lot seen by defendants. The court denied strict liability based on the plaintiff's intervention and substitution of his judgment for that of defendant experts. Courts should apply this rule—that inter-

---

(1973) (The court was following an earlier ruling in *Allied Properties v. John A. Blume & Assoc.*, 25 Cal. App. 3d 848, 102 Cal. Rptr. 259 (1972), where marine engineers were not subjected to strict liability for the design of a pier and landing floats in front of a resort hotel). *See also* La Rossa v. Scientific Design Co., 402 F.2d 937 (3d Cir. 1968). There was an additional factor of state interest supporting a similar denial of strict liability in *Hall v. New York*, 106 Misc. 2d 860, 435 N.Y.S.2d 663 (N.Y. Ct. Cl. 1981). *But see* Broyles v. Brown Eng'g Co., 275 Ala. 35, 151 So. 2d 767 (1963) (finding an implied warranty of fitness for design of drainage plans).

237. 40 Cal. App. 3d 573, 115 Cal. Rptr. 99 (1974).

vention of human judgment negates liability for human experts—to AI or ES.

The gap between the inhuman capacity for tracing logical inferences, balancing assumptions and chains of reasoning, calculating interacting probabilities, and comprehension of the ways of the human world known as "common sense," explains most of the differences between the finely-reasoned and inapposite determinations of the finest expert systems. For example, compare an ES capable of laying out the best possible course of medical treatment for the dead patient, and the direct call to the undertaker placed by the human observer. It is precisely this interposition of human judgment that also breaks the chain of liability between the manufacturer and the user of a program. This quality—the quintessential exemplification of reasonableness that the law refuses to distill from the multifariousness seen in raw jury verdicts, knowing it is incapable of delimitation—justifies imposition or denial of liability. The core of the entire legal process depends upon application of liability to the human shoulders best able to bear responsibility for providing this feature. If the user is more directly responsible for the causation of the injury than the manufacturer, imposition of liability simply cannot be justified.

Support of this rule can be obtained by comparing the reasoning and decision in *Fredericks v. Associated Indemnity Corp.*[238] with its successor case, *McMillan v. Fireman's Fund.*[239] In each case, the plaintiff pleaded that the defendant unreasonably terminated her disability benefits, and that, due to human failure to review the program properly, the computer ceased issuing the necessary checks because the program directing the insurance failed to issue the order. In *Fredericks*, the court held that the failure was unreasonable and imposed liability,[240] while in *McMillan*, the court held that the failure was not unreasonable and denied liability.[241] In *Fredericks*, however, the termination arose from computer error, and in the end it was the absence of human judgment which made the difference in liability.[242]

The need to balance justice, by imposing liability only where the opportunity to apply human judgment exists, applies to AI or ES pro-

---

238. 401 So. 2d 575 (La. Ct. App. 1981)

239. 448 So. 2d 899 (La. Ct. App. 1984).

240. *Fredericks*, 401 So. 2d at 578-79.

241. *McMillan*, 448 So. 2d at 900.

242. *Fredericks*, 401 So. 2d at 578. Abdication of decision-making in human affairs, allowing supplanting of human judgment by computer judgment, led one court to assess violation of civil rights and issue an injunction requiring human judgment over bill disputes before terminating any services. *See* Palmer v. Columbia Gas of Ohio, Inc., 342 F. Supp. 241 (N.D. Ohio 1972), *aff'd sub nom.* Palmer & Taylor v. Columbia Gas of Ohio, Inc., 479 F.2d 153 (6th Cir. 1973).

grams at least as much as it does in corporate liability cases. *Petition of Kinsman Transit Co.*[243] is an instructive, but factually complex, case that delineates some of the fine distinctions. Both the imputed knowledge of the mooring of a corporate barge—the inadequate mooring resulted in collisions, collapse of a bridge, and the partial flooding of Buffalo—and the assessment of liability collapsed when the court determined that individuals sufficiently high in the corporate hierarchy correctly assigned the task to those with superior confidence but with inferior, and from the point of view of imputed knowledge, insufficient, status. The court noted:

> [Davies'] knowledge is imputed to the corporation on the issue of exoneration, but that is precisely what the statute forbids on the issue of limitation. . . . The query seems especially pertinent when, as here, there is every indication that nothing different would have been done if George Steinbrenner had been on the scene during the final mooring as he had entrusted the operation to one admitted more competent to oversee it than he was.[244]

Judge Andrew's dissenting statement in *Palsgraf v. Long Island R.R.*[245] is quite applicable here: "It is all a question of expediency, . . . of fair judgment, always keeping in mind the fact that we endeavor to make a rule in each case that will be practical and in keeping with the general understanding of mankind."[246]

At least one court recognizes that computer operations, by themselves, do not accord with human judgment such that a legally binding determination is made. In *Travelers Indemnity Co. v. Fields*,[247] the court upheld the jury determination that where no human prevented the computer from automatically cashing a post-accident, late policy payment, the company had, nevertheless, not accepted the payment such that it was required to continue coverage. This is in direct contrast to *State Farm Mutual Insurance Co. v. Bockhorst*,[248] where the agent had the ability to intercept the check and failed. Here the court stated a paradigm for AI liability cases that may be with us for the next twenty years: "If the computer does not think like a man, it is man's fault."[249] With regard to the crucial question of which particular man is to blame, *Bockhorst* focused on the last individuals able to impose judgment: "The reinstatement of Bockhorst's policy was the direct result of the errors and oversights of State Farm's human agents and employ-

---

243. 338 F.2d 708 (2d Cir. 1964).

244. *Id.* at 715.

245. 248 N.Y. 339, 162 N.E. 99 (1928).

246. *Id.* at 354-55, 162 N.E. at 104 (1928) (Andrews, J., dissenting).

247. 317 N.W.2d 176, 185-87 (Sup. Ct. Iowa 1982).

248. 453 F.2d 533, 536-37 (10th Cir. 1972).

249. *Id.* at 535-37.

ees."[250] Accordingly, each particular case must consider who or what made the decision or took the action that caused the injury; any intervening human judgment will snap the causal chain sufficient to justify imposition of strict liability for the defective service.[251]

The following three key elements must be considered in every service liability case involving defective AI or ES applications:

(1) The class of defect, not merely the individual programming or interpretive error, must be such that it is correctable or preventible for the majority of those potentially affected;

(2) The nature of the service performed by the AI or ES must be essentially a mechanical situational response rather than one requiring creative or innovative response; and

(3) The injury must arise directly from the action or advice of the AI or ES with no actual intervening human judgment (or the opportunity for judgment willfully or negligently foregone).

Undoubtedly, a number of individual pragmatic concerns not considered here will arise in future cases, and will affect the development of this branch of law as the cases come before appellate courts. Still, between policy and pragmatic considerations, what conclusions can be drawn about service liability against an allegedly defective AI or ES application developer or manufacturer?

D. CONCLUSIONS AND EXAMPLES

The easiest of the four examples to consider is the AI-guided factory. This is almost certain to require design and implementation on a hand-crafted rather than on a mass-produced basis. The differences in local layouts, production requirements, and delivery methods will ensure that individual circumstances drive the design and implementation. Without the mass market quality, strict liability is inappropriate.

Individual subportions of the AI application, however, may qualify if their nature differs. The factory builder may subcontract a general class of AI applications that internal moving robots use for timing, signalling, and avoidance measures. Here, the mass market element is provided or potentially provided; since the factory developer voluntarily undertakes the interchange, and the interpretation is one of definable types (masses, velocities, cost qualities), few policy reasons for denying strict liability exist. The individual pragmatic circumstances that give

---

250. *Id.*

251. For example, in *International Paper Co. v. Farrar*, 102 N.M. 739, 700 P.2d 642 (1985), a box manufacturer was sued for shipping boxes that were not fit for their intended purposes. Even where the manufacturer used an ES shipping advisor, the error could have resulted from either inaccurate computer calculation, improper human input, or improper human reasoning. The court left this question of fact unanswered.

rise to the injury must be assessed carefully. If the injury reflects a decision that injury to one human being is necessary to prevent a catastrophic danger, or is a consequence of unforeseeable action by a possibly panic-stricken or otherwise uncontrollable human,[252] strict liability may not be appropriate. Only creative human judgment could be expected to resolve the dilemma presented, without some loss or injury. On the other hand, in a situation where human override can prevent harm to one person without catastrophic results, then strict liability might be appropriate, because human presence or intervention should be reasonably certain. In the event a human makes any overriding judgment, a chain of strict liability is broken that otherwise might have been forged.

The analysis of strict service liability for the environmental monitor AI application is a more difficult case. Here the mass market element is assumed, which demolishes most of the argument against liability contained in the previous example. Although each individual's desires or environment differs, clear broad classifications can be drawn (number of rooms, temperature and humidity control problems, security measures desired, contacts with outside agencies such as police and fire departments to be established). A system that fails to prevent freezing pipes in Minnesota winters would not have the right to claim unique environmental circumstances or unpredictable domain. The services provided cannot be considered judgmental but rather of a situational response nature, hardly more complex (theoretically) than the actions of a thermostat; thus no element of professionalism is involved. Because purchase of a system is voluntary, market factors and strict liability properly function as guarantors of minimal quality. A defense challenging causation is likely to be sufficient to block the cause of action only if the injury arose from human judgment overriding the AI monitor's actions. Here, more than anywhere else, the service liability cause of action seems well adapted to the societal framework.

In the pharmaceutical expert system application, the mass market element still exists, but the possibility that human judgment will be necessary now intrudes. Although the service is identical for many distinguishable classes of customers, the pharmacist must observe and assess the prescription purchaser's condition in order to properly apply the ES advice. ES use is still voluntary on the prescriber's part and the domain of knowledge is definable, but the domain of effect is still uncertain. No general rule of applicability seems to be acceptable; rather,

---

252. Consider a suddenly ill human. The collision between an Aeromexico jetliner and small plane over Cerritos, California, in 1986, was partially the result of the small plane's intrusion into controlled airspace. The pilot of the small plane apparently suffered a heart attack and lost control of his flight. *See* Malnic, *The Aeromexico Diaster—Gathering Clues in the Search for Answers*, L.A. Times, Sept. 7, 1986, § A, at 3, col. 1.

courts must accept or refuse this cause of action based on all of the policy and pragmatic considerations examined below.

Finally, the financial planning expert system application seems to give rise to a cause of action for service liability. The mass market and distinguishable classes exist, as does the certainty of the underlying domain—mathematical manipulation of economic representations. Certainly, no-one must use such a system; access is voluntary and therefore the market can balance strict liability concerns. However, any developer or manufacturer can challenge and defeat a strict liability claim when the injured party acted on his own interpretation of the terms, rules, or results of the consultation. This is also true where an injured party relied on results that even human experts could not guarantee because of uncertainty regarding the underlying assumptions (e.g., a certain rate of profit or immunity from taxation). Since human experts, if challenged, could deny even the lesser causes of malpractice or negligence under such circumstances, the ES developer or marketer should not be subjected to a higher standard.

If strict liability is an unavailable cause of action because of an uncertain underlying assumption or an interpretation that required human judgment, then what other cause of action exists which could enforce a higher standard than simple negligence? One theory, malpractice, provides some of strict liability's protection in enforcing a minimum standard that is higher than the norm. This theory balances reasonable expectations from a higher viewpoint or norm, that of the average layperson. The next section examines the potential for bringing a cause of action for malpractice on the part of the AI or ES developer or marketer.

V.   MALPRACTICE: CURRENTLY UNAVAILABLE, POTENTIALLY APPLICABLE

A.   THE CAUSE OF ACTION

Malpractice imposes liability for negligence judged by a higher standard than that of a reasonable man; it judges according to the standard of a profession's average practitioner.[253] Malpractice thus serves as a midpoint between strict liability and negligence. The base standard is set by the professional community.[254] The plaintiff is guaranteed that

---

253. Magrine v. Krasnica, 94 N.J. Super. 228, 240, 227 A.2d 539, 546 (1967) ("The vast body of malpractice law, presumably an expression of the public policy involved in this area of health care, imposes upon a dentist or physician liability only for negligent performance of his services—negligent deviation from the standards of his profession.").

254. See, e.g., Plutshack v. University of Minn. Hosp., 316 N.W.2d 1 (Minn. 1982). The issue of whether the community is local, national, or international presently is changing in the medical profession and is not within the scope of this article.

performance below the average collective judgment of knowledgeable practitioners will be recompensed. However, before any finding of malpractice is possible, the minimum standards must be established in the law.

One analyst, considering whether programmers in general may be subjected to malpractice suits, stated: "Malpractice, which typically applies to lawyers, doctors, accountants, and architects, is a statutorily created theory of liability. As such, express statutory language which includes data processing vendors is a necessary prerequisite to the maintenance of any malpractice action against the data processing vendor."[255]

Traditionally, this cause of action has been restricted to professions which require judgment and independent analysis—domains where superior human thought is considered necessary. The principle exemplar domains are medicine and law; engineering is a recent addition. Since the purpose of AI and ES programs is to provide, duplicate, or support both reasoning and judgment,[256] the question can be posed whether any basic techniques, interfaces, procedures, or checks are so essential that systems without them automatically are deficient.[257]

If express statutory language is a prerequisite, this cause of action will not be available for some time. To date, both federal administrative law and one court have determined that programming is not a profession[258]—at least with regard to requirements for overtime pay under the Fair Labor Standards Act.[259] Inasmuch as all AI and ES are built upon other programs, represent the most advanced area of practice, are scarcely a handful of years out of research laboratories, and are changing too fast for any practitioner to trace, the overall field cannot be called a "profession" at this stage. Instead, it is in mere infancy. However, the field and practice are growing so rapidly that they may require some evolution of the common law into this general area to protect the public. Can or should there be a cause of action for malpractice against those who create and sell an AI or ES program?

255. Cristo, *The Applicability of Negligence and Malpractice to Data Processing Situations*, 1 COMPUTER L. REP. 570, 570 (1983).

256. Since the bulk of ES programs, to date, are aimed at assisting professional practitioners, plaintiffs who wish to implead as many potential defendants as possible undoubtedly will consider alleging this and other causes of action against ES manufacturers.

257. These might include I/O filters, explanation features that permit human examination and verification of both reasoning and supporting fact base for conclusions, tracing capacity for correction or updating of the knowledge base, and user-verification before alteration is permitted. *See* B. BUCHANAN, *supra* note 2.

258. *See* 29 C.F.R. § 541.302(h) (1987) (administrative law); Pezillo v. General Tel. & Elec. Information Sys., Inc., 414 F. Supp. 1257 (M.D. Tenn. 1976), *aff'd*, 572 F.2d 1189 (6th Cir. 1978).

259. 29 U.S.C. § 213 (1938).

B. CURRENT STATUS AND NEAR-TERM PROSPECTS

In *Triangle Underwriters, Inc. v. Honeywell, Inc.*,[260] while the
court did not declare invalid two causes of action for computer malprac-
tice (somewhat more specifically delimited), it did indicate a general
disapproval of such claims.[261] This disinclination towards the creation
of a new tort was echoed three months later in *Chatlos Systems, Inc. v.
National Cash Register Corp.*:[262]

> The novel concept of a new tort called "computer malpractice" is pre-
> mised upon a theory of elevated responsibility on the part of those who
> render computer sales and service. Plaintiff equates the sale and serv-
> icing of computer systems with established theories of professional
> malpractice. Simply because an activity is technically complex and im-
> portant to the business community does not mean that greater poten-
> tial liability must attach. In the absence of sound precedential
> authority, the Court declines the invitation to create a new tort.[263]

In both cases, plaintiffs retained at least one cause of action to pursue
against defendant.[264] A court may have to make a harder decision
when no other pleaded cause of action is applicable. Therefore, the
plaintiff's best recourse, at present, is to pursue the general raft of alle-
gations including strict liability, malpractice, fraud, and negligence
whenever grounds for such claims exist.

A litigant faces very grave difficulties in attempting to apply a mal-
practice cause of action against an AI or ES creator. First, no legislative
or otherwise established standard for the underlying profession of
programmer or computer scientist exists. Second, the field seems too
young to expect the profession of "knowledge engineer" to be anything
more than a definition.[265] The field is far from having sufficient, gener-

---

260. Triangle Underwriters, Inc. v. Honeywell, Inc., 604 F.2d 737 (2d Cir. 1979), *aff'd*,
651 F.2d 132 (2d Cir. 1981).

261. Triangle's complaint included allegations of "failure to supervise and correct defi-
ciencies in the system" and "wrongful withdrawal of support personnel," which plaintiff
characterized as computer malpractice. The appellate court affirmed dismissal of these
causes of action, finding that (1) the statute of limitations had run for all contract causes
of action; (2) to the extent that these counts alleged fraud they were restatements of the
contract claims and thus barred; (3) the refusal to extend the continuous treatment con-
cept to plaintiff was correct because there was no professional relationship (which appli-
cation of the doctrine in any context requires); and (4) since the essence of the contract
(for a turn-key system of hardware and software together) was one for goods, rather than
services (which had a sufficiently long statute of limitations), these and all negligence
claims were also time-barred. *Id.* at 744-46.

262. 479 F. Supp. 738 (D.N.J. 1979), *aff'd*, 635 F.2d 1081 (3d Cir. 1980), *cert. denied*, 457
U.S. 1112 (1982).

263. *Id.* at 740 n.1.

264. The plaintiff's cause of action in *Triangle Underwriters* sounded in fraud in the
inducement; The cause in *Chatlos* sounded in warranty.

265. This term is generally attributed to Dr. Feigenbaum at the Heuristics Program-

ally-agreed-upon practices and basic techniques, let alone training or educational programs. Third, the field is changing so rapidly that any standard considered could be outmoded within a year or two.[266] Fourth, courts cannot easily identify the proper profession within which malpractice should be evaluated. A flaw might arise from the province of the programmers, the knowledge engineers, or the domain professionals who provided the knowledge base.[267] If the source of the injury arises in a domain which, in itself, is not subject to malpractice, then interposition of an AI or ES application should not give rise to the action.[268]

At present, creation of any AI or ES system in a domain that can serve as a basis for a malpractice action, requires the work of a team coordinating their expertise—the programmers, knowledge engineers, and domain experts. Furthermore, statisticians and reviewers are needed to evaluate and test the result. These experts will evaluate, update, and translate the knowledge basis. Translation encounters multiple difficulties when further communication with the user is required— the user may not know or comprehend the limitations on the denotations and connotations of the interface language. These problems are being addressed at present, but their solutions are not yet the subject of general knowledge.

---

ming Project, Computer Science Department, Stanford University. *See* BUILDING ES, *supra* note 2, at 12-28.

266. Five years after graduation from a first-class university, average computer science graduates must re-educate themselves in the advances that took place in the interim.

267. *Cf.* Magrine v. Krasnica, 94 N.J. Super. 228, 227 A.2d 539 (1967) (court declined to hold the dentist liable for the manufacturer's fault, because the domain of expertise is different). The court concentrated on the best source of knowledge of how to correct the underlying flaw, not the method whereby the flaw is transmitted to harm the plaintiff; this has been the guiding rationale behind the insistence on uncovering the essence of the relationship.

268. Despite the arguments adduced by some authors, it appears unlikely that any court, in the near future, would consider failure to use an expert system to be malpractice. *But see* Willick, *Professional Malpractice and the Unauthorized Practice of Professionals: Some Legal and Ethical Aspects of the Use of Computers as Decision-Aids*, in COMPUTING POWER, *supra* note 102; Petras & Scarpelli, *Computers, Medical Malpractice, and the Ghost of the T.J. Hooper*, 5 RUTGERS J. COMPUTERS & L. 15 (1975). The physician will never have all the possible inferences drawn by the program checked against the literature (the search space would be much too large). The physician thus has no knowledge of how well informed the system is, until it has been checked by other experts. However, proper testing design has been a major problem. The physician may not have the information needed for correct inferences by the program if it is interactive. In such circumstances, neither the physician nor the computer has the information that the pair "has" in a subjunctive sense. Moreover, the physician operates with a sense of resource limitations (time, money, and risk factors for tests) which, in itself, may require yet a different program to assess (and there can be a host of unforeseen difficulties in their interactions).

C. EXTENDED PROSPECTS

The practice, as well as the mathematical basis, of computer science is expanding in our society today. "Expanding" should be read in two ways: (1) the reach and extent of applications are snowballing into more and more domains and households, and (2) the commonalities and necessity for standardization to minimize avoidable conflicts are spreading throughout the community.[269] Although barely out of research labs, AI and ES programs will be subject to this process of cultural assimilation. Even as society has adjusted to other technological advances, from automobiles[270] to radios[271] to radar, these new tools assuredly will become part of the background of our societal environment. Along with this expansion and assimilation comes establishment of basic knowledge and principles (without a common support, expansion is impossible). Will such "basics" exist for AI and ES programs? What form should, or will, they take?

The decision in *Helling v. Carey*[272] indicates that failure to perform a test is negligent whenever: (1) the consequences are potentially grave and devastating, (2) the test to disclose a problem is simple, relatively inexpensive, and of such a nature that in interpreting its results there is not a judgment factor involved, and, finally, (3) giving the test is not only harmless, but is also conclusive. This negligence, in turn, translates to malpractice.

These requirements can reasonably be translated into computer terms. If a definable, specifiable, Boolean test exists for a necessary condition to successful and safe operation of an AI or ES program—a condition such that if not met, all reasonable practitioners would see consequential action presenting an unreasonable risk of grave and devastating harm—and the test can be implemented with little cost in computer accuracy, performance, or programming expertise, then failure to include such a test would constitute malpractice. Furthermore, if any action exists which is required across all domains, regardless of circumstances, failure to include such an action could be seen as malpractice.

D. CONCLUSIONS

The malpractice theory, as it currently exists, can not be easily ap-

---

269. One "guesstimate" is that the transfer rate downward from graduate to undergraduate study or from one educational level to the next is approximately 3% per year. *See* OH! PASCAL!, *supra* note 15, at 25. In leading universities, 10% is closer to the truth.

270. *See* Henningson v. Bloomfield Motors, Inc., 32 N.J. 358, 161 A.2d 69 (1960); MacPherson v. Buick Motor Co., 217 N.Y. 382, 111 N.E. 1050 (1916).

271. *See* Stevens v. Seacoast Co., 414 F.2d 1032 (5th Cir. 1969); The T.J. Hooper, 60 F.2d 737 (2d Cir. 1932).

272. 83 Wash. 2d 514, 519 P.2d 981 (1974).

plied to the four hypothetical examples used above because neither programming nor the domain for each example (with one exception) is considered a profession. Thus, the initial requirement is not met. Beyond this, however, individual considerations allow some distinctions to be drawn as to future applicability of this cause of action.

In the case of the automated chemical factory, the AI's failure to ensure that boundary conditions of the factory's environment mesh (such as events at or beyond the factory's physical location) might be considered adequate grounds for malpractice.[273] Regardless of the system's complexity, each system must account, in some way, for the existence of the world outside. Similarly, an AI or ES lacking the capability for authorized human override to allow emergency escapes would be suitable, although the potential for catastrophic damage to other workers or to the neighborhood (as in the Union Carbide disaster in Bhopal, India) would create serious problems with allowing or disallowing an override. (In the above-discussed death of the Japanese worker, the lack of any halting switch on the on-coming repair machine might be considered an omission giving rise to this cause of action.) These problems must be identified, faced, and solved before any consideration of professional, rather than experimental, standards can be undertaken. Until that time, a malpractice cause of action should be denied.

For the home environmental monitor, the ability to recognize power failures or fires might be seen as sufficiently basic and universal to justify the imposition of strict liability. Although the implementation depends upon the professional's skills and craftsmanship, any layman is aware of the hazard's existence. Other hazards and interactions are more complex. Fire and burglar alarms and countermeasures must be balanced against the risk of losses from false alerts, while any voice-activated system must cope with the uncertainties of natural speech. (How unprofessional is a failure to recognize the owner's speech when the owner is suffering from a severe cold or sore throat, regardless of the seriousness of the failure to respond?) Again, the appearance and proliferation of these systems in the market will establish the pragmatic experience and standards from which, eventually, some professional status may be derived. Until that time, the malpractice cause of action lacks its necessary prerequisite.

Finally, for both ES examples (the pharmaceutical interaction monitor and the financial advisor), developers may be required to provide for some verification of key input data to limit the impact of mistypes when any rule or process-altering adaption is made. Developers must

---

273. This is subject to the limits of discrimination, of course. If one rule governs chemical processes at temperatures above 100°C, the second ought to cover temperatures at or below 100°C.

remember the GIGO (garbage in, garbage out) rule, and consider it when designing any machine. Yet, the trade-offs between certainty and performance, between verification and speed, are real market differences and are likely to remain so. Because of the sheer lack of experience with these applications, minimal necessary precautions are speculative, and a malpractice cause of action is simply premature.

These tests or actions might be developed in the courts, although, preferably, the risk of litigation would inspire programmers, knowledge engineers, and domain experts to cautiously and thoroughly explore and implement their own internalized basic assumptions.[274] It may take decades, as was the case for previous technological advances, before the advanced technology practitioners establish the minimal underlying basics and standards. It may take decades more before these standards are solid enough to rise to a "professional" standard. Yet, this development seems to be an inevitable consequence of the specialized, abstruse, and ineluctably precise, knowledge that is involved in construction of AI or ES programs, independent of the subject domain. Until that time, however, the best cause of action for most people might be that which views the AI or ES as nothing more than the conduit whereby the reasonableness of human behavior may be evaluated: the cause of action for negligence.

VI. NEGLIGENCE: MOST APPLICABLE WHILE THE FIELD DEVELOPS

A. THE CAUSE OF ACTION

And so we come, at last, at very long last (perhaps at too long last), to the default cause of action: negligence. Negligence is an amorphous concept that is used when no specific theory may be alleged. Because the elements of negligence frequently vary according to jurisdiction and subject matter, a bewildering array of questions exist about what is, or is not, required to state in a cause of action for negligence in a given situation. For a thousand years, Anglo-Saxon law sought to look beyond the acts of implacable and impenetrable entities—gods, peers, corporations, or computers—to find the actions or omissions of individuals and to assign responsibility based on people, not on polity. This assign-

---

274. Many mistakes arise because very basic assumptions are not checked, and an unsuspecting user, who does not share the assumptions, runs afoul of them. Most medical expert systems, for example, assume that the patient is alive when they are consulted. How many financial expert systems assume that they are working with positive investments?

One way for AI and ES teams to avoid these problems is to document their basic assumptions and make these (and their grounds) available to the user. This would also permit an easier inquiry into comparative negligence.

ment led to a prolixity of formalisms and specifications defining negligence in different areas of human contact. Undoubtedly, further specification will arise with the advent of new technologies. Two theoretical differences emerge between cases involving AI or ES applications and cases involving other instrumentalities: (1) the existence and nature of the duties delimiting acceptable (and unacceptable) behavior of the defendant developer or marketer of the AI or ES; and (2) the balance of human expectations of behavior aimed at preventing harm by both the developer and user (i.e., the issue of reasonableness with regard to the application and its context).

B. THE DUTY TO ONE'S FELLOW—MAN?

How is duty defined for AI and ES programs? One analyst suggests that a case-by-case approach is required:

> Because most software must be tailored to each user, perhaps the only way to show breach of duty of care is to produce an expert witness who in fact was able to do precisely what the allegedly negligent vendor failed to do in exactly the circumstances in which the vendor failed. This would at least show that the job was technically feasible and that an expert should have been able to do it properly. If the task was technically impractical from the beginning, given the present state of the art, it is hard to see how a negligence theory could prevail, particularly if the impracticability was itself unknowable.[275]

The initial problem is that the injury by and of itself does not, and cannot, constitute proof of negligence:

> Even a correct program may fail to execute properly, or may work perfectly for a time only to fail catastrophically later without any lack of proper care on anyone's part. . . . If a program was tested for two years without finding a condition which later damages a plaintiff, and if the plaintiff can prove that a certain flaw could have been detected by testing the program for an additional twenty-four hours, would this constitute a proof of breach of duty of care? Two factors dictate a negative response. First, there is no way of knowing prior to the failure where to look for the flaw. Second, it is assumed that the state of the art would have been applied in a reasonable way to find as many potential sources as possible.[276]

---

275. Gemignani, *supra* note 15, at 190-91 (citations omitted).

276. *Id.* at 191-92 (footnotes omitted). *See also* Nycum, *supra* note 83, at 12-14. The author notes, that under the negligence test, the programmer is not an insurer. In addition, the question of comparative negligence can arise from the user's actions. Nycum seems to have assumed that a single programmer was involved, when, in fact, the situation with AI and ES is more complex. Typically, a team of individuals is involved, so no one comprehends all possible consequences. Two, or possibly three, distinct fields of expertise are involved: the domain of the expert, the translation of the knowledge engineer, and the implementation within the computer environment of the software engineer.

Determining negligence can become a dilemma when particular errors are absolutely unpredictable,[277] yet inevitable. The very process of error correction is likely to introduce additional, and more dangerous, errors.[278] Some mathematical hope (although no practical implementation to date) for formal verification exists for algorithmic programs. AI and ES, however, depend upon different reasoning techniques and are not capable (even in theory) of such provability. These programs, almost inevitably, involve reasoning techniques and implementation details that are only partially verifiable. Moreover, the range of possible interpretations (and of potential errors) of definitions comprising knowledge representation, and thus the program's model of the environment, can be unverifiable. This lack of perfectibility suggests that negligence is the most appropriate theory, at least at present. Perhaps as the field matures, programmers will develop partial verification techniques and related tools for minimizing inevitable flaws. Just as one court requires seagoing vessels to have a radio,[279] another court may require that program creators verify critical algorithmic portions of their programs.

Currently, courts have not formulated a clear definition of the duty that is owed. "Negligence as a theory for claims by third parties for personal injury or economic loss will pose practical difficulties because the "duty" owed for the design or use of computer products is undefined. Such design and use is an art."[280] Courts must first delimit the source and nature of the duty that these new applications bring into being.

1. *Duty to Supply Information With Reasonable Care*

If courts view the AI or ES as a tool for manipulation of the environment and knowledge, and the developer or manufacturer is engaged in the business of selling that tool, then the implementators may be bound to the duty stated in the *Restatement (Second) of Torts*:

> *Section 552: Information Negligently Supplied for the Guidance of Others*
>
> [O]ne who, in the course of his business, profession or employment, or in any other transaction in which he has a pecuniary interest, supplies false information for the guidance of others in their business

---

277. *Cf.* Jordan, *supra* note 99, at 6 ("[I]n light of the recognized fallibility of computers . . . it would seem that the primary negligence . . . would be attributable to the owner or operator of the dangerous device, who could reasonably have been expected to anticipate such a failure and provide adequate safeguards against harm, such as an alternative control system.").

278. *See supra* note 23 and accompanying text.

279. *See* The T.J. Hooper, 60 F.2d 737 (2d Cir. 1932).

280. Walker, *supra* note 100, at 690-91.

transactions, is subject to liability for pecuniary loss caused to them by their justifiable reliance upon the information, if he fails to exercise reasonable care or competence in obtaining or communicating the information.[281]

This definition does not seem particularly pertinent to AI, where the program primarily is devised and sold because of its ability to "do," that is, to affect the world, rather than because of its knowledge or internalized information.

The phrase "in the course of his business" was strictly construed to deny liability against the defendant computer manufacturer in *Black, Jackson and Simmons Insurance Brokerage, Inc. v. IBM*.[282] The defendant contracted with the plaintiff to sell hardware and internal software to a brokerage firm for processing information. The court held in favor of the defendant concluding that defendants were not in the business of supplying information but were selling merchandise, and the information supplied by defendant was not supplied for the guidance of plaintiff in its dealings with others. Because the AI purchaser is more interested in the program's ability to work with and change the real world than in its informational processes, this duty does not apply.[283]

However, an ES application whose sole purpose is to manipulate and transform knowledge would support a cause of action according to the *Restatement*. The fact that the ES's implementor is not directly providing the information on which the user relies does not decouple the implementor from the chain of causation, but only removes him a step. However, because privity is not even required to maintain a negligence cause of action, simply providing the tool for obtaining the information, rather than providing the information itself, should not automatically make the defendant negligent. Just as service providers and builders must act in a reasonable manner, an ES provider must

---

281. RESTATEMENT (SECOND) OF TORTS § 552 (1977).

282. 109 Ill. App. 3d 132, 440 N.E.2d 282 (1982).

283. *Cf.* Palmer v. Columbia Gas of Ohio, 342 F. Supp. 241 (N.D. Ohio 1972), *aff'd sub nom.*, Palmer & Taylor, 479 F.2d 153 (6th Cir. 1973). In *Palmer*, the court issued a preliminary injunction prohibiting Columbia Gas from terminating heating services (necessary for human life in Ohio's winter) unless it made residence service informing the customer. In the event of any billing dispute, the court prohibited action by non-managerial staff and required factual inquiry and direct individual response. The court wanted to ensure that human judgment, rather than computerized billing or non-judgmental human bureaucratic procedures, stood between the individual and the potentially life-threatening cessation of services. Thus, if an AI program existed that disconnected gas, or phone services, a lack of adequate provision for human errors or humanitarian concerns would be negligent. Human judgment, and possibly humane, rather than economic action, would be mandated by legal process.

work in a reasonable manner given practice techniques, client demands, and the marketplace balance.

This claim is supported by *Independent School District No. 454 v. Statistical Tabulating Corp.*[284] where the court allowed the school district to bring a negligence claim against the third party service bureau that (allegedly) provided inaccurate calculations of the value of a school that had burned down, leading the school district to suffer an underinsured loss. The court stated:

> [O]ne may be liable to another for providing inaccurate information which was relied on and caused economic loss, although there was no direct contractual relationship between the parties. . . . The duty to do work reasonably and in a workmanlike manner has always been imposed by law, and not by a defendant's gratuitous comments.[285]

The court considered factors including: the existence, if any, of a guarantee of correctness; the defendant's knowledge that the plaintiff would rely on the information; the restriction of potential liability to a small group; the absence of proof of any correction once discovered; the undesirability of requiring an innocent party to carry the burden of another's professional mistakes; and, the promotion of cautionary techniques among the informational (tool) providers.[286]

Courts found negligence in several cases involving the manufacturer of commercial airline maps, where the incorrect maps were at least a partial cause of the crash. In *Aetna Casualty & Surety Co. v. Jeppesen & Co.*,[287] the Ninth Circuit did not challenge the trial court's finding of defective use of different scales for the profile and plot views, but it did find error in the trial court's failure to consider the crew's comparative negligence in using the information (available in both numeric and graphic representations), and reversed the case in favor of the plaintiffs. Then, in *Saloomey v. Jeppesen & Co.*,[288] the Second Circuit affirmed jury verdicts of liability for defective, mass-produced data contained in the charts. Finally, in *Fluor Corporation v. Jeppesen & Co.*,[289] the California court held the defendant liable for inaccurate information, even though the defendant was required to place the government-provided information into its charts. Although the *Fluor* ruling may give rise to a duty,[290] litigants will view the ruling skeptically if it is used to claim that any faulty information can serve as a basis for a negligence cause of action. When improper input may lead to a result

---

284. 359 F. Supp. 1095 (N.D. Ill. 1973).

285. *Id.* at 1097-98.

286. *See id.* at 1098 (citing Roeny v. Marnul, 43 Ill. 2d 54, 250 N.E.2d 656 (1969)).

287. 642 F.2d 339 (9th Cir. 1981).

288. 707 F.2d 671 (2d Cir. 1983).

289. 170 Cal. App. 3d 468, 216 Cal. Rptr. 68 (1985).

290. *See supra* text accompanying notes 275-281.

that is clearly hazardous to the human user, either by virtue of the user's reason, or as documented by the manufacturer, then a duty of human conscientiousness is invoked.

2. *The Duty to Provide For Reasonable Handling of Unreasonable Inputs*

In *Neal v. United States*,[291] the court did more than take judicial notice of the "garbage in, garbage out" rule; Judge Biunno also stated a duty to protect against spurious errors:

> The computer is a marvelous device that can perform countless tasks at high speed and low cost, but it must be used with care. This is because it can also make errors at high speed. Those who use computers for record and accounting purposes, including the government, are accordingly obliged to operate them with suitable controls to safeguard the reliability and accuracy of the information.[292]

Improper inputs can produce immensely hazardous consequences for any AI or ES system, although the ES systems have a built-in filter in human interpretation which may catch completely impossible answers in time. The AI or ES developer can only partially control the inputs themselves. Although the developer can specify exactly the language or the source of the input that will be recognized (an AI-guided robot platform without sound sensors cannot hear any warning noises), the developer cannot specify, or even predict, what the input itself will be. Some basic programming techniques and algorithmic procedures are used in every AI or ES to cope with standard human or real world interactions and to catch standard input errors such as number transpositions.[293] These techniques, however, slow processing time and trade

---

291. Neal v. United States, 402 F. Supp. 678 (D.N.J. 1975).

292. *Id.* at 680.

293. In *Clements Auto Co. v. Service Bureau Corp.*, 298 F. Supp. 115 (D. Minn. 1969), the court noted that there was no check built into the computer to control or catch numbers or names in which characters were transposed, despite the existence of self-checking, error-controlling algorithms. *See supra* note 132 and accompanying text. A computer can easily check for such errors by asking the clerk to either repeat or confirm the input, or both, for more important situations. The cycle of input checking must come to an end, however, if any processing is to occur, just as the cycle of offer and acceptance must end for a contract to exist. Social patterns and the need for a firm rule may influence judicial logic and impose a simple requirement. For example, in standard situations, double-checking the input is all that is necessary; after that, the input is "in the mailbox," or legitimately in the processing queue. Hence, responsibility for consequential failure rests with the "sender," not the AI or ES application "recipient." Thus, any queue processor should test for an empty queue before attempting to take an item out; any stack filler should test to see if the stack limit has been reached before trying to fill it; and any single-character input should be tested for a match against the acceptable responses; if only "y" and "n" are acceptable for "yes" and "no," typing a "q" or a carriage return should be accounted for. It would not be reasonable to expect a computer ordinarily to expect the

efficiency for security. Increases in checks, reviews, and verification reduce actual production within the same time span. The trier of fact must therefore review the reasonableness of the design process, the utility and risk of the design, and the feasibility of safer alternatives.[294] Courts must exercise their judgment to determine when input, error-trapping algorithms are necessary and when they can be foregone. It must also be remembered that "unreasonable" is not always determined by what a human might view as unreasonable. At least one court noted that the machine's limitations must be considered. In *City of Lincoln v. Bud Moore, Inc.*,[295] a computer failed to notice that defendant's dramatic 'decrease' in electric bills was caused by an increase in usage past the 9,999 units the 4-place meter could count. The court did not find this to be 'unreasonable' behavior for the computer.

The easier the provision of a particular safeguard, the more likely the creator will be found negligent. When providing a safeguard becomes cheap, and its effect upon the class of possible plaintiffs (and each member thereof) becomes negligible, and its existence becomes well known, omission of the safeguard will constitute negligence. If *Helling v. Carey*[296] is any indication, AI and ES developers probably have an affirmative duty to provide relatively inexpensive, harmless, and simple, input error-checking techniques. Determining whether the AI or ES developer or manufacturer has met this duty depends on the specific factual circumstances of each case. When products from toasters to auto brakes already contain microchips, and when the price of memory and performance is cut in half every five years, judges and juries will soon be forced to make those factual determinations.[297] The details will have to await jury verdicts, which might include assessments of comparative negligence. All that can be said with certainty is that the cause of action will exist and failure to provide any safeguards will be negligent in light of inevitable input error.

---

user to sever the input cord, pour a soda into the electric keyboard, or swiftly kick the side of the CPU.

294. *See* Stanley v. Schiavi Mobile Homes Inc., 462 A.2d 1144 (Me. 1983).

295. 210 Neb. 2d 647, 316 N.W.2d 590 (1982).

296. 83 Wash. 2d 514, 519 P.2d 981 (1974).

297. Courts may create special exceptions to the loose, general concept of negligence where the input can be directly life-threatening. Such examples include the use of an AI program that: (1) administers medicine to a patient, (2) performs potentially explosive chemical manufacturing, or (3) engages in moving heavy materials across a populated yard in a factory. References might then be made to the concept of ultrahazardous activities and instrumentalities. *See* RESTATEMENT (SECOND) OF TORTS § 520-524 (1977). If an AI or ES program user complains because of performance vis-à-vis a non-using competitor, courts may refer to the "participants exception." *Id.* § 523.

3. *The Duty With Regard to Unknown Weaknesses or Flaws*

Courts acknowledge that negligent designs are possible. In *Cronin v. J.B.E. Olson Corp.*,[298] the court noted: "A defect may emerge from the mind of the designer as well as from the hand of the workman."[299] Programs can fail because their design did not account for real world complexity. This can be either a design flaw, an imperfect specification, or an imperfect understanding of the real world. With AI or ES, the question turns less on the result or consequence affecting the user—a result which may have been caused, in part, by the user himself—and turns more on how well the developer considered, addressed, and then implemented solutions to real world problems.

Generally, the AI or ES developer must prove that, on balance, the benefit obtained through the design chosen for the program outweighs the inherent risk of danger.[300] In California, the developer must show that the balance was reasonable according to the state of the art at the time of design.[301] The focus is on "the reasonableness of the manufacturer's actions in designing and selling the article as he did."[302] For example, the choice of an imperfect or potentially flawed AI or ES application might be considered negligent simply because the risk of failure was incorporated into the design, if an algorithmic solution were possible but omitted because it was economically infeasible or undesirable. However, if AI or ES implementation included use of inference techniques that were imperfect or potentially incorrect (e.g., if there were a depth cutoff in the search process or a time-out in the reasoning process, designed to favor efficiency over absolute certainty), and the purchaser or user had notice of this, then the user could not claim that such a failure is negligent. A developer should only promise that the system will give the user the best answer the system is capable of giving; the developer should not promise perfection, which may be unobtainable in any event.

A developer's failure to warn that there may be imperfect results could give rise to a cause of action for negligence. A duty certainly exists to warn of known flaws.[303] This duty extends beyond the time that the AI or ES is shipped to the purchaser, though how far it might extend is subject to multiple factors. Thus, while an AI or ES developer may not be required to fix known flaws—particularly if the repair may

---

298. 8 Cal. 3d 121, 501 P.2d 1153, 104 Cal. Rptr. 443 (1972).

299. *Id.* at 134, 501 P.2d at 1162, 104 Cal. Rptr. at 452.

300. *See* Barker v. Lull Eng'g Co., Inc., 20 Cal. 2d 413, 426-27, 573 P.2d 443, 457, 143 Cal. Rptr. 225, 233 (1978).

301. *See* Balido v. Improved Mach., Inc., 29 Cal. App. 3d 633, 640 (1972).

302. Roach v. Kononen, 269 Or. 457, 465, 525 P.2d 125, 129 (1974).

303. *See* Kammer v. Lambs-Grays Harbor Co., 55 Or. App. 557, 639 P.2d 649 (1982). *See also* Johnson v. William C. Ellis & Sons Iron Works, Inc., 604 F.2d 950 (5th Cir. 1979).

be more hazardous than the occurrence—the developer still may be required to inform all customers or users of the existence of the flaws and ways to avoid or diminish the effects as they are uncovered. Failure to meet this duty could give rise to an action for negligence.[304]

While an AI application involves interaction with the unpredictable real world, an ES is limited to the domain(s) in which it operates. Because the ES has a narrower focus and range of possible interpretations than an AI, there is an extension of predictability within that range.[305] Both systems include some modelling of how they fit within a lesser world—whether internally stated or simply exemplified by the program. Foreseeability thus includes all of the traditional problems inherent in any negligence action, plus a new concern: How reasonable was the designer's and user's interaction with the AI or ES, and through it, with each other? A jury must decide whether some presumption of human capacity and judgment existed (or should have existed) in the AI or ES, i.e., whether the application depended upon interaction with a reasonable user of ordinary sensibility (RUOOS). This presumption gains importance when input may be available to the user that is not available to the program, or when the information flow is not a two-way street.

C. REASONABLE BEHAVIOR AND THE BALANCE OF EXPECTATIONS

If negligence arises from imbalanced expectations concerning behavior of people and programs, then where might the balance lie? What should determine this?

A user may not know how the AI or ES is designed to reason; yet the user can expect that the computer lacks knowledge of the user's real goals and common sense. However, expectations of non-cognoscenti cannot be the only basis for a claim any more than with other complex areas of human endeavor. As the judge in *Hoven v. Kelble*[306] noted:

> Members of the medical profession who have held themselves out to be supermen should not be surprised that laymen take them at their word and impose superdamages. The law, however, recognizes the medical profession for what it is—a class of fallible men, some of whom are unusually well-qualified and expert and some of whom are not. The stan-

---

304. *Cf.* Arizona Highway Dept. v. Bechtold, 105 Ariz. 125, 460 P.2d 179 (1969) (judgment for the motorist based on the highway department's negligence for failure to have preventive maintenance of computer monitored and generated signals).

305. With a given reasoning capacity, whether human or machine, a more narrow search can be deeper. Analogous reasoning—which can be emulated in AI or ES programs—leads to the conclusion that the designers can focus on the *use* of interpretations more if they are less concerned with *possible* interpretations.

306. 79 Wis. 2d 444, 256 N.W.2d 379 (1977).

dard to which they must conform, however, is determined by the practices of neither the very best nor the worst of the class. Like automobile drivers, engineers, common laborers, and lawyers, they are obliged to conform to reasonable care in the circumstances.[307]

1. *How Can We Establish a Standard?*

There is much hyperbole in nontechnical literature about what AI or ES can do both now and in the immediate future. The promotional statements of marketing staff may create expectations that present techniques and present developers cannot meet. Because the marketing staff is not better able, nor better qualified, than the user to establish the balance of expectations, its statements should count as mere puffery, which the user is reasonably expected to discount. However, as with any complex field in which a jury decides, the people with knowledge of the field (the practitioners) are able to argue effectively for, or against, a given standard's reasonableness. Unfortunately, as long as a practitioner's assertions are not so devoid of sensibility as to call for judicial intervention, the practitioners will be free to influence the jury's understanding of "reasonable." However, it should not be forgotten that the question of "reasonable" is ultimately one for the jury.[308]

2. *Design Negligence: What Is Feasible at a Given Time Sets the Standard For Each Case*

An obvious starting point for setting up minimum standards for developers' duties and AI and ES performance levels would be to establish statutory guidelines. However, no statutory standards exist at this time. Further, creating statutory standards does not seem wise because any such stagnant pronouncement would ill-serve this dynamic, developing field in comparison to the flexible and adaptable standards available through jury judgments. Sacrificing the community's common sense in favor of a rigid, albeit more predictable, dogmatic assertion of justice is more likely to lead to injustice than to serve the present and future needs of the community.

Statutory standards may fail because real world complexity defies specification or prevention. The statutory minimum negligence standard could declare that it would not be reasonable to expect better behavior from another human being with identical support in predictable, algorithmic computation facility, given a situation similar to the one

---

307. *Id.* at 457, 256 N.W.2d at 385 (citation omitted).

308. *See supra* note 295. *Cf.* Lopez v. Wolf Mach. Co., 133 Cal. App. 3d 324, 183 Cal. Rptr. 695 (1982) (the insured's declaration that his beliefs regarding coverage were reasonable did not create an issue of fact).

where the program failed.[309] Another type of failure is the mismatching of specifications in the program design. If the specifications of the AI or ES call for different standards of accuracy within the algorithms used by the program, failure to account for the resulting potential problems is negligent design.[310]

If the choice between menus, mice, and mistyping as input tools is a marketing point (inasmuch as the designs may be equivalent from a mathematical standpoint), the absence of solid evidence that one method or another is more apt to produce failures indicates that the difference cannot serve as grounds for alleging negligence. However, if one input source is proven more prone to produce errors in the program than another, this could establish design negligence on the part of the developer.

> *In Balido v. Improved Machinery, Inc.,*[311] the court stated:

> Although separate counts for negligence, warranty and strict liability have been pleaded, we view them as stating a single cause of action, in that the complaint seeks damages for personal injuries caused by deficiencies in the design of a manufactured product. . . . [T]he manufacturer is not an insurer of the safety of its product, and the test for strict liability is the same as that for negligence, except for the element of scienter. Strict liability for deficient design of a product (as differentiated from defective manufacture or defective composition) is premised on a finding that the product was unreasonably dangerous for its intended use, and in turn, the unreasonableness of the danger must necessarily be derived from the state of the art at the time of design. A danger is unreasonable when it is foreseeable, and the manufacturer's ability, actual, constructive, or potential, to forestall unreasonable danger is the measure of its duty in the design of its product.[312]

In *Balido*, the defendant conceded that, because the design violated a pre-existing statutory requirement, it was deficient. The court went on to consider the issues of lapse of time and whether a superceding cause arose from the purchasing company's failure to buy inexpensive, available safety features, despite repeated warnings by defendant about the

---

309. A second analyst reaches the same conclusion with regard to algorithmic computer programs. *See* Gemignani, *supra* note 15.

310. This type of error has occurred. The Vancouver Stock Exchange index was calculated to four decimal places, but truncated (not rounded) to three. It was recomputed with each trade (some 3000 each day) resulting in an average loss of an index point a day. Three weeks of work by consultants from Toronto and California computed the proper corrections for twenty-two months of compounded error. While on Friday, November 25, 1983, the index stood at 524.811, it began Monday morning at 1098.892 (up 574.081) while the stock prices were unchanged. *See* Lilley, *Vancouver Stock Index has Rate Number at Last*, Toronto Star, Nov. 29, 1983, § C, at 11, col. 9.

311. 29 Cal. App. 3d 633, 105 Cal. Rptr. 890 (1972).

312. *Id.* at 640, 105 Cal. Rptr. at 895 (citations omitted).

safety deficiency.[313]

Even if the manufacturer feasibly could have made the product safer ("feasibly" denoting a risk/benefit analysis), liability is not established by simply showing that a better mousetrap could be built.[314] There is the basic problem of perfect hindsight obscuring imperfect foresight, which is particularly acute when contemplating a bug in a program. The entire range of a program, especially a learning, artificially intelligent, or expert system program, often is beyond the predictive ability of the humans who create it. Yet the mistakes are often simplistic in the extreme.[315] A court will find negligence for a design error when the failure's source is either an unreasonable design or specification error, which leads to the absence of a necessary process, necessary data in the knowledge base, or source of knowledge for verification.[316] At present, there is no known way to incorporate common sense about the world into a program that even approaches human capacity. Therefore, AI reasonably cannot be expected to behave in human fashion over the entire range of possible inputs reacting through its feedback loop. Common sense is a commodity desired in AI just as much as it is desired in naturally intelligent individuals, whether experts or laymen. But common sense in programs cannot be expected at present. The very name *artificial* intelligence indicates that these programs are not yet equivalent to humans. It would be unjust to allow a human to take advantage of the potential holes in AI comprehension and claim wrongful injury as a consequence.[317]

---

313. *Id.* at 641-46, 105 Cal. Rptr. at 896-99; *cf.* Stanley v. Schiavi Mobile Homes, 462 A.2d 1144 (Sup. Ct. Me. 1983) (proof of unreasonable danger involved examining the utility of the design, the risk of the design and the feasibility of safer alternatives).

314. *See* Brady v. Melody Homes Mfr., 121 Ariz. 253, 589 P.2d 896 (1978), *rev'd in part*, Dart v. Wieve Mfg., Inc., 147 Ariz. 242, 709 P.2d 876 (1985) (But note: the jury still must evaluate why the developer did not build the better mousetrap).

315. A punctuation error led to the loss of the Mariner probe to Venus. In a FORTRAN program, DO 3 I = 1,3 was mistyped as DO 3 I = 1.3. This was accepted by the compiler as a legal syntax and semantics. It assigned 1.3 to the variable DO 3 I, instead of performing 3 repetitions of some process. *See Introduction*, 6 ANNALS OF THE HISTORY OF COMPUTING 6 (1984).

316. Design error would have been applicable when the splashdown of the Gemini V orbiter missed its landing point by 100 miles because its guidance program ignored the motion of the earth around the sun. *See* J. FOX, SOFTWARE AND ITS DEVELOPMENTS 187-88 (1982). *See also supra* note 14 and accompanying text.

317. In *Travelers Indemnity Co. v. Fields*, 317 N.W.2d 176 (Sup. Ct. Iowa 1982), the defendant mailed in his premium check several months late and after he had been involved in an accident. The plaintiff insurance company had already issued many warnings and opportunities to renew. The plaintiff's first computer cashed the check, while the second computer issued a refund check later the same day; the refund check was based on an expired policy. The court held that the insurance company's acceptance of the defendant's check constituted a waiver of the delay, or a reinstatement of his policy. *See also* McMillon v. Old Republic Life Ins. Co., 33 Ill. App. 3d 658, 342 N.E.2d 246 (1975) (involv-

Within a given domain, however, for the program to be considered "expert," the expert's basic sense must be replicated in the behavior of the program's outputs. This expertise may, and indeed probably will, change over time.[318] The knowledge engineer's failure to incorporate a mechanism to allow for this change cannot be considered a negligent act inasmuch as it only reflects an alteration to the time of implementation. However, there may be times when the knowledge engineer would be negligent if the rule lacked any, or any reasonably accessible, means for altering the ES to account for changes in the domain.[319] If the knowledge engineer incorporated an explanation facility, or included a means for the ES to find the outside source which validated the underlying knowledge base, the engineer may or may not be considered negligent. If the knowledge engineer adds a validation technique which allows a program to perform at, or above, the level of experts in the domain (according to testing procedures that simply evaluate the outputs),[320] and the program user has less expertise than the engineer, then little value will be gained by enabling the user to evaluate the program's reasoning process through an explanation facility. The program's reasoning may use sophisticated techniques, such as multiple tiers of mathematical evaluation of fractional probabilities, that the user would find incomprehensible. Generally, the lack of an explanation facility (although generally useful for debugging, improvement, or making the ES attractive to users) should be considered negligent only if the comparison to human judgment reasonably could be expected to trap errors.

3. *A Knowledge Flaw Must Be Judged by the Domain Standard*

Design negligence affecting algorithmic choices or representation techniques can be separated, in theory, from flaws arising in the choice of knowledge, explanations, or actions used or taken by the AI or ES. These choices are much more the result of the domain expert's (or program purchaser's) specifications.

Some medical domain ES operate from a statistical basis concerning prior and consequential probabilities, using data gained from large-scale

---

ing a special Illinois statute; held insurer had not accepted payment where machine automatically deposited check but promptly issued refund).

318. For example, the value of performing bypass surgery, rather than using drugs to treat cardiac problems, may alter as knowledge of the domain changes, but the program may not be kept up to date and thus could offer a newly inaccurate conclusion.

319. Considering the constant changes in tax law over the past eight years (at least three overhauls and the 1986 reform), any tax-related ES that did not allow for frequent alteration would arguably be negligently designed, particularly if there were little or no documentation or warning of its dated correctness.

320. For example, consider the double-blind diagnostic results comparison that has been performed for MYCIN. *See* MYCIN, *supra* note 67, ch. 30 (*The Problem of Evaluation*, Ch. 31 (*An Evaluation of MYCIN's Advice*).

technical studies. The location, date, and methodology of the study affect the reported data—data which forms the basic source for the ES's knowledge base. If an error results from this data's incompatibility with the realities affecting the injured party, the flaw lies not with the ES, but with the expertise from which it operated. The people who provide the initial data have the greatest incentive and ability to prevent such flaws. As suggested earlier,[321] a plaintiff could sue on a negligence cause of action for improper provision of information. Even if statutory mandates or procedural restrictions imposed the form of representation, the negligence action lies against the information provider for the form of information used in the system.[322] However, courts must consider the time and process whereby the information was entered into the knowledge base. So long as the information was reasonable for the applicable time and conditions when it was entered, then there can be no grounds for recovery. Evaluation of the reasonableness of the knowledge base can and should require a jury decision.

Dealing with an AI or ES that challenges (if not overwhelms) human capacity to reach the correct answer (i.e., that performs better than human capacity can emulate) can be problematic. If a flaw results from an unforeseen interaction with the knowledge base—a flaw undetected by any human before it causes an error, than a court may find it difficult to assess negligence against the knowledge provider. Where knowledge providers did not perceive the problem when the information was provided,[323] they can be seen as serving the knowledge base, rather than its extensions produced by the program.

4. Common Sense Failure—The User's Standard

A factory worker in Kobe, Japan, entered a restricted zone where machinery was in operation, began to fix one of the machines, and became quite wrapped up in his work—fatally, as he failed to notice or avoid the approach of a delivery machine.[324] The delivery machine did not stop because its operation was governed by a program that was not programmed to respond to such an emergency. This may have been the

---

321. *See supra* text accompanying notes 281-290.

322. For a discussion on the tort liability of ES contributions, see Willick, *Professional Malpractice and the Unauthorized Practice of Professionals: Some Legal and Ethical Aspects of the Use of Computers as Decision-Aids*, in COMPUTING POWER, *supra* note 102, at 845-47.

323. This is a current problem in AI which challenges one of the basic assumptions often used: that the computer can know all of the logical implications, or theorems, of its base set of axioms. How do you reason with any incompletely known field and still avoid inconsistencies that would invalidate all future inferences? *See* Fagin & Halpern, *supra* note 43.

324. Lohr, *New in Japan: The Manless Factory*, N.Y. Times, Dec. 13, 1981, § 3, at 27, col. 1.

first AI-caused fatality, and it appears to have occurred due to the worker's negligence. There is a somber warning here: the average individual is too prone to ascribe common sense to computers.

What is the just determination when an injured plaintiff claims that the developer was negligent in failing to prescribe for unforeseen circumstances brought about by human shortcomings or exaggerated expectations? The plaintiff may have placed too much reliance on the computers, so that plaintiff's own essential human traits of judgment, intuition, and abstract reasoning were negligently underemployed. Of course, lack of common sense is not a commodity desired in AI or ES any more than it is desired in naturally intelligent individuals, whether experts or non-experts. However, it is precisely these elements of common sense and human understanding that pose the most baffling problems to researchers in these fields.

A system purchaser should make a preliminary assessment to determine: (1) what cautionary measures, if any, the developer implemented in order to warn the AI or ES user about the limitations of the programs;[325] and (2) what features of the input and output devices are patently observable. When a discrepancy exists between successive outputs that is analogous to a basic limitation in the AI's or ES's input, then the human user, whose skill in analogical reasoning is presently far superior to any computer's, cannot be too surprised by any error. If developers will clearly state the assumptions and basic principles from which the AI or ES system operates and create an easy method for the user to observe the reasoning process used by the AI or ES system, then developers would give human users the ability to assess the program's success or failure to meet the desired goals (which may, in fact, be different than the specific goals, but this difference may not show up without experience).

Ascription of human sensibility to AI programs is erroneous for the following reasons:

(1) Most people have multiple senses constantly operating, which serve as independent sources of information. This interdependence allows for better judgment than commitment to a single source (vision, sound, or gauge), which is all many AI have at present;

(2) Most people have years of learning and cultural experience which they incorporate into their judgment; most AI programs are closer to idiot savant status—superb in only one area, childlike in any other, and lacking comprehension of the context; and

---

325. *Cf.* Ward v. Hobart Mfg. Co., 450 F.2d 1176 (5th Cir. 1971); Balido v. Improved Mach., Inc., 29 Cal. App. 3d 633, 105 Cal. Rptr. 890 (1972); Kammer v. Lamb-Grays Harbor Co., 55 Or. App. 557, 639 P.2d 649 (1982). These cases involve a manufacturer's duty to warn.

(3) Most people have access to multiple means to resolve doubts, conflicting truths, or other problems, such as other people or higher authority. AI has little internalized in the way of these consistency-correction techniques and so acts on the conclusions.

Courts must insist that expectations of human performance from inanimate machinery is unreasonable, now, and for the foreseeable future. Human judgment, senses, and adaptability far outperform the most sophisticated AI in the wide variety of environments in which people must operate. As a consequence, whenever AI interacts with people, the latter have the burden of presuming a lack of capacity beyond that directly observed (not inferred). For ES, however, courts may insist upon a higher standard—behavior or performance meeting the standard which is the norm for the domain in which the ES operates.[326] Reasonableness, a slippery and flexible standard, defies a single definition that applies to all situations, domains, and environments; it is the quintessence of a jury's evaluation of negligence.

D. CONCLUSION AND EXAMPLES

A host of remaining issues are important for any negligence case: proximate cause, intervening cause, comparative negligence, foreseeability, misuse, or damages. However, these analyses are very fact specific, and thus, will not be discussed at length here.[327]

AI and ES applications are the intervening connection between the creator's efforts to transmit human expertise and multiply human value over a general class of situations in a fashion that unavoidably creates an incomplete and imperfect specification of the infinitely complex real world. Because of the efforts of users or affected persons to cope with the specific details of the problem and the context in which they find themselves, these novel, complex creations are, in effect, the first tools developed for interactive transmission of human intelligence and judgment. Both sides share an equal burden to use reasonable care to iden-

---

326. This will vary according to the basic standard for the domain. An ES in medical pathology would be expected to have the accuracy of its M.D. peers; an ES in financial advising would be expected to have the accuracy of its non-professional peers.

327. It is interesting to note that, no matter how unpredictably precedent may operate, and no matter how uncertain the usable boundaries of the analogical reasoning—which is the strength of both the common law and the human mind, and is presently unrepeatable by mechanical entities—it nonetheless works.

L. Thorne McCarty and his associates have been working at Rutgers University on legal reasoning and artificial intelligence. Other researchers include: Edwina Rissland and Kevin Ashley at the University of Massachusetts; James A. Sprowl at Northwestern University; and Anne Gardner at Stanford University. Layman Allen has led them all with his erudite essays on logic and law at the University of Michigan. Compliments to each and to those unnamed due to space limitations.

tify and avoid mistakes, errors, confusion, and disasters. This burden cannot be thrust callously upon the uncaring (and unhuman) machine.

Different results occur when AI and ES are applied to the four hypothetical cases stated above.[328] When the actions and interactions of the machinery on the floor of an AI-run chemical factory are laid out, and a working schedule for the production line is established by the constraints of natural law,[329] then human individuals must not divert from their assigned tasks in the blithe assumption that the AI will handle their unforeseen interruption with human judgment. At the same time, it is unreasonable for developers to make an assumption of perfection or uninterrupted successful completion of each step of the program. In the infinitude of rules and consequences, Murphy's Law is undeniable. A developer must not create an AI that is unreasonably optimistic in design, such as a processing plant that is impervious to situation-specific override through authorized human judgment. Every worker need not be able to stray outside his safe areas without assuming a risk, but someone, whether a ground supervisor or manager, must have the capability to halt or otherwise interrupt processing should an emergency occur. Given the unforeseeable complexity of the real world, a design lacking the capacity to interweave human judgment is clearly unreasonable.

The home-monitoring AI must assume a great deal of change in the rules and conditions. Where the conditions deal more with comfort than with health, the actions and constraints of the system would change and interchange with the human judgment. When the action or situation may be life-threatening, however, then the ability to provide authorized human judgment overriding the system's determination becomes critical. Any uncontrolled fire requires human attention, as would any perceived internal flaw of the monitoring system itself. But, if one vent serves two rooms, neither the inability to keep them at the different temperatures nor a conflict between automatic energy-saving light manipulation and insomnical demands for brilliance to drive the night away, would be an unreasonable flaw in the AI monitor.

The ES monitoring pharmaceutical interactions cannot be expected to gauge when a user is lying about illegal drug use. However, the developer cannot reasonably assume that there is no need to consider such an interaction and to leave it out of the knowledge base. These sub-

---

328. *See supra* § III(D).

329. The AI system requires an ordered scheme of reactions, with a consequential scheduling of material shipments. Temperature constraints for the various reactions, as well as time delays for completion, will be unavoidable, just as the production line imposes task orders and time limits on the (formerly human) work stations. These constraints are dictated by the task, not by who (or in this case, what) performs the action.

stances are an unpleasant reality permeating the society for which the ES is designed.

The ES financial planner must be expected to assume that the user is providing it with true data unless it uncovers inconsistencies, such as a depreciation below basis plus improvements or different valuations entered for the same item purchased by one individual. Similarly, the developer cannot reasonably exclude particulars concerning penalties, interest, and fees. There are times when tax violations are either unavoidable (for reasons separate from tax considerations and thus beyond the range of the ES) or reasonable (for reasons inherent in tax code oddities or other personal financial problems such as a divorce).

Negligence, as a cause of action, allows the most direct evaluation of the affected individual's behavior. In this sense, description of the nature of the AI or ES, as designed by the creators and perceived by the users, is equal to the context-specific details that caused the injury. The description then is nothing more than a complete specification to the plaintiff, defendant, and jury, of the entire world as the human actors found it. This forces both parties to a human standard—it neither forces perfect behavior nor permits the abandonment of one's own judgment. The standard remains quintessentially human: did each party act reasonably? Acting reasonably will be the hardest capacity for the AI or ES to emulate, if it ever can be done.

AUTHOR'S NOTE

When I started this Article, I had only one presumption: no single theory of liability would justifiably cover all of the possible interactions involving human injury which might occur due to AI or ES applications. I did not know whether a new theory might be needed, and wanted to retain an open-minded approach in light of the potential for change. As I narrowed the theories and applicable causes of action down to the four considered, partially because of the desire to limit the size of a constantly growing Article, I grew more satisfied with my initial presumption. Each theory was unacceptable for some reason: products liability, because a large class of problems exist that are not susceptible to predictable, verifiable results; service liability, because the exemptions for professional practice and potential for predictable classes of problems made current case constraints both too lax and too strict; malpractice, because there always will be an inherent conflict between finding the computer knowledgeable within a constricted domain yet ignorant of human affairs—bedeviling the chance to define a "professional" performance; and negligence, because there are classes of problems where a strict cause of liability should be imposed upon those able to prevent harms as superior knowledge accumulates or professionally-actuated

minimal standards evolve. None of these flaws, however, justified creation of an entire new theory in my mind. The adaptive ability of the common law seemed more than adequate, given a proper understanding of the underlying field of computer science and its subdomains, artificial intelligence and knowledge representation.

These subdomains are, at most, in adolescence. We can expect a horde of new products, new businesses, and new practitioners intent on making artificial intelligence and expert systems commonplace in our lives, if only because they can provide applications concerning the twin certainties of death and taxes. Estate planning programs, tax advising programs, actuarial evaluation programs, stock-buying programs, widget manufacturing programs, and virtually anything that can be done on a regular basis might be reduced to an application program. How well these attempts succeed or fail will influence how soon the consequences show up in the courts. When the case arises, facts are alleged, and evidence sought, the attorneys and courts must reflect on what it is that they are doing—bringing the best in human judgment to bear in a novel situation. Let us hope it is done well. Ought we hope that justice will or will not itself be reduced to an applications program, edifying the promise and the threat of justice that is blind?

AI and ES applications are new tools in our society. The law is an old, yet constantly-renewed, tool for adjusting our society. It is my sincere hope that this Article may help the latter assimilate the former.