Tytuł: **Rola teorii obliczalności w badaniach nad AI**

Autor: Piotr Kołodziejczyk; pkolodziejczyk@interia.pl

Źródło: http://kognitywistyka.net; mjkasperski@kognitywistyka.net

0. Uwagi wstępne

Celem mojego artykułu jest próba uchwycenia związków pomiędzy matematyczną teorią obliczalności a badaniami nad sztuczną inteligencją. Podejmuję zatem próbę odpowiedzi na pytanie o to, czy włączenie teorii obliczalności w ramy AI jest koniecznym warunkiem prowadzenia badań nad sztuczną inteligencją. W przypadku, gdy AI utożsamia się wyłącznie z działalnością informatyczno-inżynierską, której cel stanowi konstrukcja systemów nieodróżnialnych od człowieka pod względem charakteru dokonywanych operacji poznawczych, moje analizy nie spełniają żadnej doniosłej roli. Jest tak, ponieważ ten sposób rozumienia zadań Sztucznej Inteligencji *implicite* zakłada instrumentalistyczne (operacjonistyczne) podejście metodologiczne, zgodnie z którym o wartości poznawczej tych systemów nie decyduje teoria leżąca u podstaw ich konstrukcji, lecz efekt ich działania. Jeżeli jednak teorię Sztucznej Inteligencji zdefiniuje się jako pewną teorię umysłu, to eksplikacja zagadnień oscylujących wokół pojęcia obliczalności jawi się wtedy jako jeden z węzłowych problemów. Rozstrzygnięcie podstawowych kwestii związanych z tym terminem zezwala bowiem na ustosunkowanie się do obecnych we współczesnych koncepcjach umysłu sporów między eksternalizmem a internalizmem semantycznym, czy dyskusji o sposób istnienia postaw propozycjonalnych – by wymienić tylko najważniejsze spośród problemów nurtujących teoretyków umysłu. Na tej podstawie sądzę więc, że krytyczna analiza koncepcji obliczalności sformułowanej w ramach AI nie jest przedsięwzięciem teoretycznie jałowym.

1. Doniosłość badań nad obliczalnością

Trywialne będzie w tym miejscu stwierdzenie, iż w ramach teorii Sztucznej Inteligencji za paradygamatyczne przyjmuje się twierdzenie o obliczeniowym charakterze procesów poznawczych systemów przetwarzających informacje. Jednakże, w badaniach nad AI zakres stosowalności terminu *obliczalność*[1] jest nierzadko nie doprecyzowany, pomimo iż obliczalność wydaje się stanowić jedną z podstawowych kategorii.

> Znaczna część badań z zakresu nauk kognitywnych – pisze David J. Chalmers – skupia się na pojęciu obliczania. Jednak wiele kwestii związanych z tym terminem wciąż pozostaje niejasnych. Czym dla obiektu fizycznego jest realizacja obliczania? Czy realizowanie odpowiedniego obliczania wystarcza do posiadania umysłu? Czy każdy system poznawczy posiada podłoże obliczeniowe? To tylko część najbardziej kontrowersyjnych pytań w zakresie podstaw nauk kognitywnych.[2]

Jest widocznym, że kwestia zasadności koncepcji umysłu sformułowanej w ramach teorii Sztucznej Inteligencji wiąże się z pytaniem o prawomocność samej teorii obliczalności. Skoro teoria ta stanowi podstawę analiz dokonywanych przez teoretyków AI, winna się ona cechować niesprzecznością i pełnością. Postawienie teorii obliczalności wymogu tego rodzaju implikuje konieczność rozważenia tych jej rozstrzygnięć, które uznaje się za fundament koncepcji umysłu zaproponowanych w ramach teorii Sztucznej Inteligencji.

Pytanie o to, czym jest obliczalność, pierwotnie wiąże się z badaniami nad podstawami matematyki. Mówiąc więc o zagadnieniu obliczalności w ramach teorii Sztucznej Inteligencji, nie sposób uniknąć rozważań na temat rozstrzygalności[3] i dowodzenia w systemach formalnych. Może się bowiem wydawać, iż

> dla każdej teorii rozstrzygalnej można zbudować maszynę sprawdzającą formuły tej teorii. Takie proste maszyny do sprawdzania wzorów z rachunku zdań były nawet budowane. Jednakże sprawdzenie pamięciowe formuł z rachunku zdań jest tak łatwe, że budowanie maszyn w tym celu jest przy obecnych zastosowaniach logiki zbędne. Ale np. maszyny rozstrzygające problemy z elementarnej arytmetyki liczb rzeczywistych, która jest również teorią rozstrzygalną mogą mieć duże znaczenie praktyczne.[4]

Wydaje się zatem, że wzajemny związek pojęć obliczalności, rozstrzygalności oraz dowodzenia, zdaje się mieć fundamentalne znaczenie dla badań nad Sztuczną Inteligencją.

---

[1] Za genezę zainteresowania matematyków pojęciem obliczalności uznaje się powszechnie wystąpienie Dawida Hilberta, który na Międzynarodowym Kongresie Matematycznym w 1900 r. przedstawił listę 23 najważniejszych nierozstrzygniętych problemów matematycznych. „Jeszcze może ważniejsze od samych problemów – postuluje Lynn A. Steen – było wyznanie wiary Hilberta, że w matematyce nie może być miejsca na *ignorabimus*. Argumentacja Hilberta – a w istocie także przykład całego jego życia – świadczyły, że w naturze matematyki leży formułowanie i rozwiązywanie zagadnień; nie istnieje więc możliwość, według sądu Hilberta, aby coś pozostało na zawsze nieznane. Narzędzi samej tylko myśli, w umysłach twórczych matematyków, powinny wystarczyć do rozwiązania każdego problemu matematycznego. Dla uzasadnienia swej tezy Hilbert (...) wziął na warsztat program kodyfikacji i formalizacji procesu matematycznego dowodu. Miał on wszelkie powody wierzyć, że formalizacja wprowadzi do matematyki tę samą pewność, jaką dwa wieki wcześniej dały mechanice prawa Newtona. Podobnie jednak jak mechanika kwantowa zburzyła determinizm newtona, tak też wynik o *nierozstrzygalności*, jaki uzyskał (...) Kurt Gődel zniszczył pewność Hilberta. (...) Gődel udowodnił (...), że – używając słów Hilberta – w matematyce zawsze istnieje pewne ignorabimus"; L. A. Steen, *Matematyka dzisiaj*, tłum. B. Osuchowska, W. M. Turski, ss. 13-23, w: *Matematyka współczesna*, red. L. A. Steen, WNT, Warszawa 1983, s. 18.

[2] D. J. Chalmers, *On Implementing a Computation*, ss. 391-402, w: „Minds and Machines", N 4/1995, s. 391.

[3] Klasę zagadnień jest rozstrzygalna wtedy, gdy istnieje metoda pozwalająca ma rozwiązanie każdego zagadnienia będącego elementem tej klasy za pomocą skończonej liczby kroków.

[4] A. Grzegorczyk, *Zagadnienie rozstrzygalności*, PWN, Warszawa 1957, s. 65.

Zakładając, że "dowód dla dowolnej formuły *A* z języka teorii jest skończonym ciągiem formuł tej teorii takim, że każda z formuł tego ciągu jest albo aksjomatem specyficznym albo aksjomatem logicznym, lub powstaje z wcześniejszych formuł w tym ciągu poprzez zastosowanie reguł inferencyjnych"[5], okazuje się, że teoretycy będący zwolennikami mocnej wersji AI *implicite* przyjmują, iż problemy analizowane przez „inteligentne" systemy przetwarzania informacji są problemami rozstrzygalnymi. Skoro bowiem każdą operację poznawczą można sprowadzić (lub zdefiniować) do zbioru (za pomocą) skończonej liczby jednostek syntaktycznych wraz z zachodzącymi między nimi relacjami, to operacja ta ma - co oczywiste – charakter obliczeniowy. Każdy problem natury poznawczej można więc rozwiązać za pomocą skończonej ilości operacji. W ujęciu tym, procesy poznawcze sprowadzają się wyłącznie do interakcji pomiędzy dyskretnymi syntaktycznymi stanami danego systemu wyrażalnymi za pomocą modeli matematycznych[6].

W świetle powyższych stwierdzeń jest widoczne, że systemy, w ramach których wszystkie problemy są rozstrzygalne są modelami finitystycznymi. Przez analogię do idei finityzmu w filozofii matematyki[7] o systemach tych można stwierdzić co następuje:

1. ich składową, i zarazem przedmiotem badań, są skończenie syntaktycznie dane struktury.

   > Poza takimi strukturami – piszą Alan Newell i Herbert Simon – system zawiera także zbiór procedur przeprowadzających operacje na wyrażeniach, aby wytworzyć inne wyrażenia. (...) Fizyczny system symboli jest w związku z tym maszyną generującą ewoluujące zbiory struktur symbolicznych[8],

2. operacje na tych strukturach mają charakter kombinatoryczny, a przez to efektywny w rozumieniu matematycznej metody efektywnej[9],

3. pojęcia abstrakcyjne, takie jak np. dowolnej jednostki syntaktycznej, czy nieskończonego zbioru symboli, nie są ani przedmiotem badań, ani elementem tego modelu.

W związku z powyższym, można przypuszczać, że klasa problemów rozważanych w ramach badań nad Sztuczną Inteligencją jest klasą problemów efektywnie rozwiązywalnych. Oznacza to, że teoretycy AI milcząco zakładają, iż każdy problem jest zupełnie rozwiązywalny za pomocą metod algorytmicznych. Jeżeli problemy te utożsamia się z analizą i symulacją procesów poznawczych, to jest również jasnym, że każdy proces poznawczy podlega procedurom algorytmizacji. Mówiąc inaczej, każdy proces poznawczy jest opisywalny za pomocą funkcji obliczalnych. Definiując funkcje obliczalne jako określone za pomocą operacji składania, rekursji[10] lub minimum efektywnego[11] można twierdzić, że analiza tych

[5] Zob. H. Rasiowa, *Wstęp do matematyki współczesnej*, PWN, Warszawa 1973, s. 250.
[6] Por. S. Bringsjord, M. Zenzen, *Cognition Is Not Computation. The Argument From Irreversibility*, ss. 285-320, w: „Synthese", N (113)/1997, s. 296.
[7] Zob. R. Murawski, *Filozofia matematyki. Zarys dziejów*, PWN, Warszawa 2001, ss. 112-113, 126-132.
[8] A. Newell, H. Simon, *Computer Science as Empirical Inquiry. Symbols and Search*, ss. 35-66, w*Philosophy, Psychology, Artificial Intelligence*, red. J. Haugeland, MIT Press, Cambridge Mass. 1981, s. 40.
[9] „Każda metoda efektywna – pisze Andrzej Grzegorczyk – jest zwykle metodą rozwiązywania nie jednego, ale całej klasy pewnych zagadnień. Na przykład, zagadnienie, czy dana liczba naturalna jest liczbą pierwszą, jest klasą złożoną ze wszystkich zagadnień tego rodzaju, jak: czy 1 jest liczbą pierwszą, czy 2 jest liczbą pierwszą, czy 3 jest liczbą pierwszą itd. Opisana (...) metoda jest wspólną metodą rozwiązywania każdego z tych zagadnień elementarnych. Ściślej więc należy mówić o rozstrzygalności lub nierozstrzygalności całych klas zagadnień"; A. Grzegorczyk, *Zagadnienie...*, s. 11.
[10] Na przykład: „Sumą $x+y$ liczb naturalnych x i y nazywamy funkcję taką, że dla dowolnego x i y jest $x + 0 = x$, $x + S(y) = S(x) + y$. Powyżej podany sposób definiowania nazywa się definiowaniem rekurencyjnym, a

funkcji (stanowiących formalną reprezentację procesów poznawczych zachodzących w danym systemie) jest badaniem klasy problemów zupełnych (*problemów typu P*)[12]. Wydaje się zatem, iż założenie o możliwości formalizacji (algorytmizacji) wszystkich procesów poznawczych implikuje przyjęcie tezy o jedno-jednoznacznej deskrypcji każdego z tych procesów. Teza ta wiąże się więc z twierdzeniem Gödla o zupełności logiki pierwszego rzędu[13]. Jeżeli bowiem formalizmy tworzące dany system są niesprzeczne (czyli z konieczności posiadają model), to należy uznać, iż są one rekurencyjnie przeliczalne. Stąd zaś można wywodzić, że klasa tych formalizmów jest zbiorem skończonym w sensie Dedekinda. Dlatego też, również i relacje zachodzące pomiędzy tymi formalizmami charakteryzuje własność skończoności. A zatem, model obejmujący formalne reprezentacje procesów poznawczych wraz z relacjami między nimi musi być modelem skończonym, co w konsekwencji prowadzi do stwierdzenia, że algorytmiczny opis natury procesów poznawczych jest opisem deterministycznym. Deterministyczne ujęcie formalnego modelu reprezentującego ogół procesów poznawczych gwarantuje więc jego jednoznaczność i zupełność, a tym samym umożliwia obliczeniowe ujęcie istoty każdego procesu poznawczego.

Podstawowa trudność związana z przedstawionym wyżej podejściem dotyczy unikania refleksji nad problemami nierozwiązywalnymi (*problemami typu NP*)[14]. Z tego więc względu, w poniższych rozważaniach koncentruje się na logicznym ujęciu problematyki obliczalności w ramach teorii Sztucznej Inteligencji. Na podstawie tej analizy (mającej głównie charakter formalny) staram się wyeksponować wnioski implikujące twierdzenie o determinacji badań nad AI przez teorię obliczalności.

2. Logiczne ujęcie problematyki obliczalności w badaniach nad sztuczną inteligencją

W ramach badań nad Sztuczną Inteligencją, analizy oscylujące wokół zagadnienia obliczalności *explicite* odwołują się do rozstrzygnięć wynikających z prac Alana Turinga, spośród których najbardziej istotną wydaje się być napisany w 1936 r. artykuł *On Computable Numbers with Aplications to Enscheidungproblem*. W nim to bowiem autor zaproponował

dokładniej mówiąc definiowaniem za pomocą schematu rekursji"; A. Mostowski, *Liczby naturalne i funkcje obliczalne* Państwowy Zakład Wydawnictw Szkolnych, Warszawa 1971, s. 62.

[11] „Mówimy – stwierdza Andrzej Mostowski – że funkcja $f(x, y)$ = *najmniejsze z takie, że* $g(x, z) = y$, określona jest za pomocą schematu minimum efektywnego dla funkcji $g(x, z) = y$, jeżeli dla każdej pary liczb naturalnych x i y istnieje liczba naturalna z taka, że $g(x, z) = y$ (...). Przykładem definiowania funkcji za pomocą minimum efektywnego jest funkcja $y - x$ określona dla $y \geq x$ następująco: $y - x$ = *najmniejsze takie z, że* $x + y = z$"; A. Mostowski, *Liczby...*, s. 122.

[12] Problemy zupełne (*P-problemy*) są grupą zagadnień, które można rozstrzygnąć za pomocą algorytmu (procedury obliczeniowej).

[13] Por. Ch. H. Papadimitriou, *Złożoność obliczeniowa*, tłum. P. Kanarek, K. Loryś, WNT, Warszawa 2002, s. 126.

[14] Pojęcie *NP-problemu* jest wynikiem „(...) podstawowych prac Stephena Cooka z University of Toronto (...). Ta klasa problemów zawiera obecnie dosłownie setki różnych problemów znanych z tego, że nie dają się rozwiązać za pomocą obliczeń"; R.L. Graham, *Kombinatoryczna teoria szeregowania*, ss. 200-228, w: L. A. Steen, *Matematyka...*, s . 213. Definiując problem nierozstrzygalny James M. Brady pisze: „Przyjmujemy, że problem jest rozstrzygalny tylko wówczas, gdy można podać pewien algorytm, który pozwala generować poprawne odpowiedzi dla każdego przypadku problemu. Dokładniej, mówimy, że problem P jest rozstrzygalny, jeśli wyznaczona przez niego funkcja całkowita f_p jest obliczalna (...). problem, który nie jest rozstrzygalny, tj. taki, że wyznaczana przez niego funkcja całkowita f_p nie jest obliczalna, nazywamy nierozstrzygalnym"; J. M. Brady, *Informatyka teoretyczna w ujęciu programistycznym*, tłum. A. Skowron, T. Kulisiewicz, WNT, Warszawa 1983, s. 71.

schemat maszyny, która zdawała się stanowić idealną konstrukcję służącą do rozwiązywania pewnych problemów językowych, a mianowicie – obliczania funkcji danych w postaci wyrażeń oraz rozpoznawania i rozstrzygania języków[15]. Innymi więc słowy, maszyna Turinga jest urządzeniem, którego zadaniem jest realizacja każdego rodzaju procedury obliczeniowej.

> Wymieńmy – pisze Stevan Harnard – to, czym okazało się być obliczanie: niezależna od interpretacji, fizycznie interpretowalna manipulacja symbolami. Tylko w ten sposób okazało się możliwe przeprowadzenie jakiejkolwiek kalkulacji, o jakiej matematycy mogli jedynie marzyć, a także spowodowanie, by maszyny wykonywały wiele z innych inteligentnych czynności, do których wcześniej zdolni byli jedynie ludzie (i zwierzęta). Prawdopodobnie było więc w tych warunkach całkiem naturalnym wnioskować, że ponieważ: (1) nie wiemy, w jaki sposób podmioty poznawcze poznają i (2) obliczanie może dokonywać tak wiele czynności, do jakich zdolny jest poznający podmiot, poznanie jest tylko formą obliczania (...). Ostatecznie, zgodnie z tezą Churcha-Turinga obliczanie jest wyjątkowe i najwyraźniej wszechmocne (...); cokolwiek mogą zrobić systemy fizyczne, mogą to także wykonać komputery.[16]

Ten sposób myślenia o obliczaniu sprawił, iż (zdaniem teoretyków AI) maszyna Turinga stanowi najbardziej adekwatny model eksplikujący naturę procesów poznawczych[17].

Formalnie maszyna Turinga jest uporządkowaną czwórką: $MT = (K, A, f, s)$, w której zakłada się, że:

1. K jest skończonym zbiorem stanów (instrukcji maszyny), zaś $s \in K$ jest stanem początkowym,
2. zbiór A jest skończonym zbiorem symboli rozumianych jako alfabet MT,
3. zbiory K oraz A są rozłączne.

Przyjmuje się również, iż:

1. zbiór A zawiera symbole specjalne (symbol pusty) i ∇ (symbol końcowy),
2. f jest funkcją przejścia odwzorowującą $K \times A$ w zbiór $(K \cup \{h, „t”, „n”\}) \times A \times \{\leftarrow, \rightarrow, \longrightarrow\}$, przy czym: h jest stanem końcowym maszyny, *„t”* – stanem akceptującym, *„n”* – stanem odrzucającym, $\leftarrow$ oraz $\rightarrow$ są kierunkami ruchu taśmy maszyny, natomiast — oznacza pozostanie taśmy maszyny w miejscu. Jest ważnym, że wymienione wyżej elementy MT nie należą do $K \cup A$ [18].

Traktując funkcję f jako program MT i wiedząc, że jest ona funkcją całkowicie określoną, można wykazać, iż obliczanie jest procesem finitystycznym. Zgodnie bowiem z tezą Churcha – Turinga, MT jest zdolna w skończonym (lecz nie dającym się precyzyjnie wyznaczyć czasie) obliczyć wartość f dla dowolnie zadanego argumentu a[19]. Znaczyłoby to zatem, że dowolna funkcja g będąca reprezentacją danego wyrażenia językowego lub stanu

[15] Por. Ch. Papadimitriou, *Złożoność...*, s. 42.
[16] S. Harnard, *Computation Is Just Interpretable Manipulation. Cognition Isn't*, ss. 379-390, w: „Minds and Machines”, N 4/1995, s. 384.
[17] Por. M. Davis, *Czym jest obliczanie?*, ss. 261-288, w: L. A. Steen (red.), *Matematyka...*, ss. 265-267.
[18] Por. Ch. Papadimitriou, *Złożoność...*, s. 37. Por. także, P. van Emde Boas, *Machine models and simulation*, ss. 1-61, w: *The Handbook of Theoretical Computer Science*, *vol. 1: Algorithms and Complexity*, red. J. van Leuwen, MIT Press, Cambridge Mass. 1990, s. 17.
[19] Pozostaje tu do rozważenia tzw. „problem stopu” dla *MT*. Kwestię ta wykracza jednakże poza ramy tego artykułu.

poznawczego jest funkcją algorytmizowalną (rozstrzygalną). Tym samym zaś – dowolne wyrażenie danego języka oraz wszystkie stany poznawcze podlegałyby procedurom algorytmicznym. Innymi słowy, dowolne wyrażenie językowe / proces poznawczy mogłoby zostać zrealizowane przez maszynę Turinga. W celu eksplikacji tej tezy zwróćmy uwagę na następujące twierdzenie:

<u>Twierdzenie 1:</u> Dla dowolnego języka formalnego istnieje *MT*, która go rozstrzyga.

<u>Założenia twierdzenia 1:</u>

1. Niech $L \subset (A — \{\ \})$ będzie językiem formalnym, czyli ciągiem jednostek syntaktycznych.

2. Niech *MT* będzie maszyną Turinga, taką że dla dowolnego wyrażenia $a \in (A — \{\ \})$:

$$(a \in L) \supset MT(a) = „t” \vee (a \notin L) \supset MT(a) = „n”.$$

Oznacza to, że w zależności od tego, czy słowo wejściowe występuje w języku *L*, maszyna Turinga generuje stan „tak” lub „nie”. W takim przypadku stwierdza się, że *MT* rozstrzyga *L*. Jeżeli więc język *L* jest rozstrzygany przez *MT*, to jest on językiem rekurencyjnym [20].

<u>Dowód (indukcyjny):</u>

1. Przyjmijmy, że $L = \{0, 1,\ \}$.

2. Dla składowych języka *L* istnieje funkcja *f* określona np. na zbiorze $R = \{0, 1\}$ taka, że zarówno dla funkcji $f(0)$, jak i $f(1)$, *MT* wygeneruje skończony ciąg o wartości ∇ *010* i zakończy działanie.

Warto w tym miejscu odnotować, iż waga podejścia Turinga dla badań nad sztuczną inteligencją polega tym, że samą maszynę Turinga można traktować jako algorytm stanowiący obliczeniową (formalną) reprezentację procesów poznawczych dowolnego systemu przetwarzającego informacje[21]. Dodatkową zaletą propozycji Turinga stanowi fakt, że *MT* jest konstruktem niezależnym od semantycznej interpretacji wprowadzanych do systemu danych. Oznacza to, iż realizacja określonej procedury jest warunkowana wyłącznie przez zaimplementowany w maszynę algorytm obliczeń.

> Systemy [skonstruowane przez] Turinga i Markowa – pisze Andrzej Skowron – mają tę własność, że znaczenie instrukcji i warunków tych systemów wystarczają na to, by przy pomocy algorytmów konstruowanych w nich, obliczane były wszystkie funkcje algorytmicznie obliczane w potocznym sensie.[22]

Jest zatem widocznym, że w świetle dotychczasowych rozstrzygnięć można powiedzieć, że

> każda sensowna próba stworzenia matematycznego modelu obliczeń komputerowych i zdefiniowania czasu jego działania musi prowadzić do modelu obliczeń i związanej z nim miary kosztu czasowego, które jest wielomianowo równoważne maszynom Turinga.[23]

[20] Por. Ch. Papadimitriou, *Złożoność...*, s. 43.
[21] Por. M. Minsky, *Computation. Finite and Infinite Machines*, Prentice Hall Inc., Engelwood Cliffs 1967, s. 123.
[22] A. Skowron, *Podstawy teorii maszyn matematycznych*, Instytut Kształcenia Nauczycieli, Warszawa 1974, s. 89.
[23] Ch. Papadirmitriou, *Złożoność...*, s. 54.

Teza ta (znana w literaturze przedmiotu jako "teza Churcha") i jest formalnie równoważna następującemu twierdzeniu:

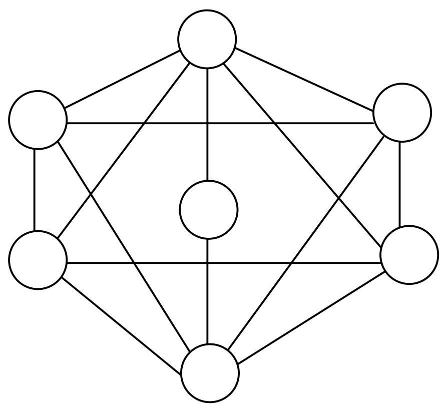
<u>Twierdzenie 2 (o izomorficzności maszyn):</u> Dwie maszyny *M* i *N* są izomorficzne wtedy i tylko wtedy, gdy działanie maszyny *M* jest wykonywane przez maszynę *N*, zaś działanie maszyny *N* jest wykonywane przez maszynę *M*. Formalnie mamy więc:

$$\exists x\ [Mx \wedge \exists y\ (Ny \wedge x = y)]$$ [24].

<u>Założenia twierdzenia 2:</u>

1. Liczba stanów maszyny *M* i *N* jest identyczna
2. jeżeli obliczenia są skończone i kończą się stanami x_M oraz y_N, to $x_M = y_N$.

Izomorfizm maszyn przedstawić można za pomocą następującego schematu:



<u>Dowód (indukcyjny):</u>

1. $M \in N$. (Na podstawie schematu można bowiem wywnioskować, że np. obliczeniu *6,0,2* w maszynie *M* odpowiada konfiguracja stanów *6,02* w maszynie *N*).
2. Pozostałe przejścia od stanu do stanu są identyczne. Zatem, maszyny *M* oraz *N* są sobie równoważne (izomorficzne). Należy więc zgodzić się ze Zdzisławem Pawlakiem, który stwierdza, iż „(...) maszyny te mimo różnej budowy mają te same możliwości obliczeniowe: to, co może zrobić jedna z nich, może również zrobić i druga"[25].

Przytoczona powyżej wypowiedź Z. Pawlaka zezwala na ukazanie bardzo ważnej konsekwencji dla teorii Sztucznej Inteligencji rozumianej jako pewna teoria umysłu. Po pierwsze: postulat głoszący, iż obliczanie stanowi podstawę modelowania i symulowania procesów poznawczych prowadzi (o czym wyżej była już mowa) do przyjęcia, w ramach badań nad AI, stanowiska funkcjonalistycznego. Po drugie zaś,

> obliczanie – pisze D. Chalmers – jest centralne w studiach nad umysłem dokładnie dlatego, że dostarcza elastycznego, prawie uniwersalnego schematu do wyrażenia i analizy organizacji przyczynowej [danego systemu].[26]

[24] Por. Z. Pawlak, *Maszyny matematyczne*, Państwowy Zakład Wydawnictw Szkolnych, Warszawa 1971, s. 37.
[25] Tamże, s. 38.
[26] D. Chalmers, *On Implementing...*, ss. 401-402.

Warto dodać, iż możliwość eksplikacji przyczynowej organizacji danego systemu za pomocą zbioru procedur obliczeniowych zakłada konieczność określenia relacji pomiędzy dokonywanym przez system obliczaniem i jego semantyczną interpretacją oraz interakcji między składnikami systemu obliczającego. Z punktu widzenia prowadzonych w tej pracy badań, pogłębionej analizy wymaga problematyka semantycznej interpretacji dokonywanych w systemie obliczeń.

3. Obliczanie jako podstawa rozstrzygnięć semantycznych

Próby wywiedzenia semantyki ze zbioru jednostek syntaktycznych (procedur obliczeniowych) są obecne właściwie od początku badań nad Sztuczną Inteligencją. Wydaje się, iż są one implikowane milcząco przyjmowanym założeniem o niemożności charakterystyki

> programu komputerowego jedynie jako posiadającego składnię bez warstwy znaczeniowej; taki pogląd jest nie do przyjęcia przez opowiadających się za komputacyjnymi modelami umysłu czy rozumienia języka naturalnego. Według nich wewnętrzne następstwa proceduralne dowolnego programu komputerowego dają mu możliwość osiągnięcia warstwy znaczeniowej.[27]

Za Jamesem Fetzerem można więc stwierdzić, że z badań nad obliczalnością prowadzonych przez teoretyków Sztucznej Inteligencji

> wyłania się pojęcie komputerów jako systemów zdolnych do wprowadzania, przechowywania, manipulowania i wyprowadzania znaków będących reprezentacjami informacji, lub mogących funkcjonować jako reprezentacje informacji (...). Haugelandowskie pojęcie komputerów jako automatycznych systemów formalnych może okazać się [w tym miejscu] najbardziej jasne, ponieważ wyraźnie uznaje on różnicę pomiędzy badaniem „egzemplarzy znaków" [*tokens*] lub „ciągów egzemplarzy znaków" jako pozbawionych znaczenia znaczników [*meaningless makers*], którymi manipuluje się zgodnie z regułami pewnej „samo- kontrolującej się gry" a badaniem ich jako znaczących ciągów posiadających pewna potencjalnie znaczącą relację do świata zewnętrznego.[28]

Stąd też można wywodzić, iż konstrukcja danej teorii semantycznej jest warunkowana zbiorem określonych reguł obliczeniowych. Ujmując to zagadnienie formalnie, uzyskujemy następujące twierdzenie:

<u>Twierdzenie 3 (o semantyczności systemów przetwarzających informacje):</u> Dowolny system przetwarzający informacje posiada własności semantyczne wtedy i tylko wtedy, gdy w ramach jego struktury istnieje taki zbiór jednostek syntaktycznych, który jest reprezentowany jako zbiór funkcji całkowicie obliczalnych.

<u>Założenia twierdzenia 3:</u>

Procedura dowodowa oparta jest o notację systemów produkcji typu BNF[29],

---

[27] J. Kloch, *Świadomość komputerów? Argument „Chińskiego Pokoju" w krytyce mocnej wersji sztucznej inteligencji według Johna Searle'a*, Wydawnictwo Biblos, Tarnów 1996, s. 96.

[28] J. H. Fetzer, *Thinking and Computing. Computers as Special Kinds of Sings*, ss. 345-364, w: „Minds and Machines", N 3/1997, s. 349. Zob. także, J. Haugeland , *Semantcs Engines. An Introduction to Mind Design*, ss. 1-34, w: *Philosophy, Psychology, Artificial Intelligence*, red. Tenże, MIT Press, Cambridge Mass. 1981, s. 22.

[29] Notacja typu BNF jest sposobem zapisu syntaktyki za pomocą zbioru produkcji Chomskyego-Posta. Oznacza to, iż notacja ta zawiera wyłącznie dwa rodzaje operatorów: symbole podstawowe oraz klasy syntaktyczne.

1. Elementami systemu posiadającego własności semantyczne są skończone ciągi symboli,

2. Dowolny system przetwarzający informacje jest funkcjonalnie równoważny działaniu dowolnej maszyny Turinga.

<u>Dowód (indukcyjny):</u>

1. Niech L będzie językiem systemu przetwarzającego informacje. Załóżmy, iż $L = \{a, b, c\}$. Przyjmijmy również, że składowe języka L są bezkontekstowe. Do systemu wprowadźmy także stałe p i k oznaczające początek oraz koniec jego pracy.

2. Stosując formalizm typu BNF możemy otrzymać następujące ciągi znaczeniowe (Z):

   $f(p) = Z(a)$,

   $f(p + 1) = Z(b)$,

   $f[(p+1) + 1] = Z(c)$,

   $f\{[p+1) + 1] + 1\} = Z(a \wedge b)$,

   $f(k)$ = STOP (KONIEC PRACY SYSTEMU).

W świetle przeprowadzonej procedury dowodowej nietrudno dostrzec, że konstrukcja systemów semantycznych na podstawie zbioru procedur obliczeniowych z konieczności implikuje tezę o internalistycznej interpretacji semantyki, w myśl której same symbole wraz z przeprowadzanymi na nich operacjami stanowią konieczny i dostateczny warunek dla przypisania dowolnemu systemowi przetwarzającemu informacje własności referencji, intencjonalności, czy zdolności do produkcji znaczeń. Akceptując wnioski płynące z tzw. *eksperymentu Jasnego Pokoju* Paula i Patricii Churchlandów[30], wielu badaczy AI *explicite* opowiada się za stwierdzeniem głoszącym, iż zaimplementowane w systemie reguły przetwarzania i operowania symbolami stanowią przyczynę występowania postaw propozycjonalnych w strukturze danego systemu[31]. Innymi więc słowy, obliczeniowe ujęcie procesów poznawczych umożliwia przeprowadzenie redukcji postaw propozycjonalnych do stanów funkcyjnych w taki sposób, iż postawy te (rozumiane jako reprezentacje mentalne) traktuje się jako wynik relacji zachodzących pomiędzy wejściami, wyjściami oraz regułami

---

Pisze J.M. Brady: „Przykładami klas syntaktycznych w formalnym języku programowania ALGOL (...) są: <deklaracja przełącznika> i <instrukcja przypisania>. (...) Przykładami symboli podstawowych w języku ALGOL są: BEGIN, : =, *. Proces generowania zdania można uważać za proces zastępowania całej struktury zdania przez szczególną konfigurację podstruktur, zastępowanie tych podstruktur przez inne itd., aż w końcu wygenerowane zostaną same symbole podstawowe"; J. M. Brady, *Informatyka...*, s . 106.

[30] Eksperyment *Jasnego Pokoju* miał na celu wykazanie, że składnia jest wystarczającym warunkiem konstrukcji semantyki w systemach AI. Pomysł Churchlandów bazował na odwołaniu się do następującej analogii fizycznej: „Wyobraźmy sobie ciemny pokój a w nim człowieka trzymającego duży magnes (...). Jeśli ten człowiek będzie wymachiwał magnesem do góry i na dół, to (...) powinno to wywołać koliste rozchodzenie się fal elektromagnetycznych, a zatem powinno się zrobić jasno. Jednak każdy z nas wie (...), że nie jest do pomyślenia, byśmy mogli uzyskiwać światło po prostu przez wymachiwanie magnesem w kółko"; P.P. Churchland, *Czy maszyna może myśleć*, ss. 17-23, w: „Świat Nauki", N 7/1991, s. 20. Analogiczna relacja, zdaniem filozofów z San Diego zachodzi pomiędzy syntaktyką a semantyką. Zob. w tej kwestii J. Kloch, *Świadomość...*, ss. 50-51.

[31] Por. P. Jacob, *What Minds Can Do? Intentionality in Non – Intentional World*, Cambridge University Press, Cambridge 1996, ss. 149-154.

transformacji charakterystycznymi dla owego systemu[32]. Odpowiadając bowiem na pytanie o warunki występowania danego stanu propozycjonalnego *F*

> staramy się wyjaśnić ten typ interakcji zachodzących pomiędzy składnikami systemu obliczeniowego, który powoduje wystąpienie stanu *F.* Ponadto, próbujemy dociec, jaka jest różnica pomiędzy obliczeniami konstytuującymi stan *F*, a klasą funkcji stanowiących warunek występowania stanów *G, H, I*.[33]

Na podstawie przytoczonego powyżej stwierdzenia V. Hardcastle twierdzić można, iż zbiór reguł obliczeniowych (algorytmów zaimplementowanych w systemie) determinuje formułowane w ramach badań nad AI rozstrzygnięcia semantyczne nie tylko w sensie formalnym, ale także i przyczynowym. Znaczy to, że zbiór formuł logicznych stanowi abstrakcyjne odwzorowanie relacji kauzalnych zachodzących w danym systemie przetwarzającym informacje, będąc zarazem przyczyną ich zachodzenia.

> Do realizacji obliczania – pisze D. Chalmers – trzeba mieć zaledwie zbiór składników, które działają na siebie przyczynowo zgodnie z pewnym wzorem. Natura składników, ani też sposób w jaki realizowane są przyczynowe połączenia między składnikami nie mają znaczenia (...). W ten sposób obliczanie jest centralne w tak odmiennych podejściach do modelowania poznania jak podejście procesów symbolicznych, koneksjonizm i sztuczne życie (...). Jeśli więc umysł może być rozumiany w terminach skończonej organizacji przyczynowej, może on być również rozumiany obliczeniowo.[34]

4. Uwagi końcowe

Podsumowując podjęte w artykule wątki można stwierdzić, że możliwość wyjaśniania organizacji umysłu za pomocą kategorii obliczeniowych z konieczności implikuje możność obliczeniowej eksplikacji jego stanów. Zatem, skoro umysł poddaje się opisowi w ramach danej teorii obliczalności, teoria ta winna także wyjaśniać takie jego własności jak: referencja i intencjonalność, czy też zdolność do generowania, przetwarzania i rozumienia znaczeń. Jest więc widocznym, iż deskrypcjom o charakterze obliczeniowym powinny podlegać semantyczne własności umysłu, czy – szerzej – dowolnego systemu przetwarzającego informacje. W związku z problematyką podjętą w tym artykule, najbardziej istotne wydają się być próby obliczeniowego ujęcia semantyki przez badaczy AI. Ocena tych prób będzie, mam nadzieję, tematem kolejnego opracowania.

[32] Zob. V. Hardcastle, *Computationalism*, ss. 303-317, w: „Cogniton”, N 3/1995, ss. 310-311.
[33] Tamże, s. 310.
[34] D. Chalmers, *On Implemeting...*, ss. 401-402.