

## **Automatyzacja i autonomia**

- \* Opiszemy automatyzację i autonomię w kategoriach połączenia czterech kluczowych aspektów: generowanie lub wykonywanie planów, działania deterministyczne lub niedeterministyczne, modele zamkniętego lub otwartego świata oraz sygnały lub symbole jako podstawowa reprezentacja wiedzy.
- \* Omówimy problem ramowy w kategoriach założenia zamkniętego świata i otwartego świata.
- \* Opiszemy ograniczoną racjonalność i jej implikacje dla projektowania inteligentnych robotów.
- \* Porównamy, w jaki sposób podejście do automatyzacji lub autonomii wpływa na styl programowania i projekt sprzętu robota, awarie funkcjonalne i błędy ludzkie.
- \* Zdefiniujemy mit substytucji i omów jego konsekwencje dla projektu.
- \* Wymienimy pięć przestrzeni handlowych w systemach człowiek-maszyna (sprawność, plany, wpływ, perspektywa i odpowiedzialność), a dla każdej przestrzeni handlowej podaj przykład potencjalnej niezamierzonej konsekwencji przy dodawaniu autonomicznych zdolności.

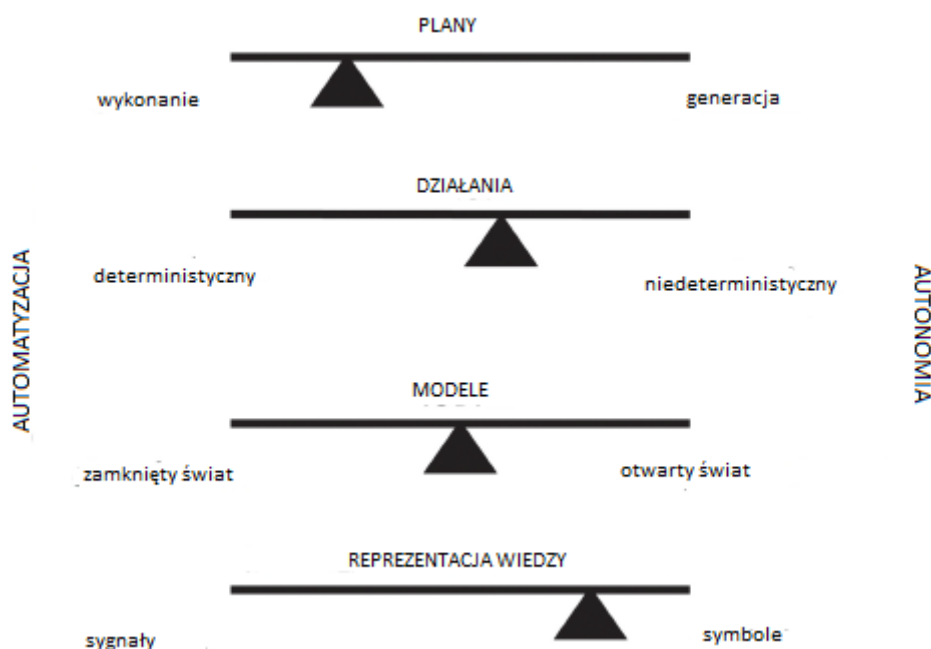
## **Przegląd**

Ta część dotyczy autonomii i tego, co ją różni od narzędzi do budowania i kontrolowania. W szczególności postaram się odpowiedzieć, jaka jest różnica między automatyzacją a autonomią? Jak widać w poprzedniej części, automatyzacja historycznie koncentrowała się na tworzeniu narzędzi do manipulacji przemysłowych. Manipulatory przemysłowe mierzyły sukces pod względem tego, jak dobrze robot może wykonywać zaplanowane, precyzyjne, powtarzalne czynności lub sekwencje. Automatyzacja wstępnie zaplanowanych ruchów manipulatora przemysłowego zazwyczaj wykorzystuje algorytmy deterministyczne i opiera się na dobrze poznanym środowisku, które jest całkowicie i precyzyjnie modelowane z wyprzedzeniem. Z drugiej strony osoby zainteresowane rozwojem robotów mobilnych mierzyły sukces w kategoriach tego, jak dobrze robot działał jako agencja; ten rozdział wprowadza ważne pojęcia w agencji. Prace badawczo-rozwojowe skoncentrowały się na robocie, który mógłby generować plany dostosowujące odpowiednie działania lub sekwencje do otoczenia na podstawie wykrytych danych. Należy podkreślić, że zdolności autonomiczne, w odróżnieniu od automatyzacji, wykorzystują niedeterministyczne algorytmy w otwartym świecie, aby przewyższyć potencjalne pominięcia i błędy w modelowaniu świata. Choć systemy niedeterministyczne są trudniejsze do przetestowania, nie oznacza to, że roboty mogą spontanicznie przekraczać swoje zaprogramowane funkcje; część wprowadza zasadę ograniczonej racjonalności, która mówi, że robot nie robi niczego, do czego nie został zaprogramowany. Często generowanie planów wymaga manipulowania symbolami (np. czerwony samochód, osoba idąca, otwarte drzwi itp.) oraz sygnałami (np. chmura punktów lidarowych, wibracja itp.), wymagających zaawansowanego przetwarzania percepcyjnego i rozumowania w celu przetłumaczenia danych z czujników w semantyczne rozumienie obiektów i czynności zachodzących na świecie. W tym rozdziale poruszono również bardziej fundamentalne pytanie: Dlaczego to ważne, że istnieje różnica między autonomią a automatyzacją? Rozróżnienie między automatyzacją a autonomią może wydawać się subtelne, ale ma poważne konsekwencje: wpływa na styl programowania robota, konstrukcję sprzętu do budowy robota oraz rodzaje awarii powodowanych przez robota. Nieprzemysłane projekty zdolności autonomicznych często powodują błędy wynikające z mitu substytucji: że robot może zastąpić człowieka bez wpływu na większą organizację społeczno-techniczną lub odporność systemu. Ponieważ ma znaczenie, że automatyzacja różni się od autonomii, ta determinacja rodzi pytania: Jakie są zalety autonomii nad automatyzacją? Czy możesz mi powiedzieć, kiedy użyć jednego, a kiedy drugiego? Ile potrzebuję autonomii? Święty Graal systemów bezzałogowych to taksonomia lub system

oceny, który można wykorzystać do określenia „właściwej ilości” autonomii dla konkretnego robota i aplikacji. Niestety, projektowanie robotów jest tylko tym: problemem projektowym ze złożonymi kompromisami. W większości przypadków sztuczna inteligencja oferuje techniki przewyższania negatywnych aspektów kompromisu, ale te autonomiczne możliwości muszą być włączone do projektu systemu od samego początku. Zasadniczo projektant musi wziąć pod uwagę autonomiczne zdolności potrzebne do wykonania misji, następnie rozważyć dodatkowe autonomiczne zdolności potrzebne do zapewnienia, że robot naprawdę wykona misję, a na koniec upewnić się, że wymagane czujniki i procesory są włączone do projektu platformy.

### Cztery suwaki autonomicznych zdolności

Zamiast próbować zdefiniować konkretną zdolność robota jako „automatyzację” lub „autonomię”, bardziej przydatne może być myślenie o zdolności jako czerpaniu ze spektrum technik reprezentujących cztery kluczowe aspekty: plany, działania, modele i reprezentacja wiedzy. Jednym ze sposobów wizualizacji zdolności jest zestaw suwaków w spektrum dla każdego aspektu pokazanego na rysunku



Należy zauważyć, że każdy z aspektów związanych z automatyzacją i autonomią wpływa na ogólną zdolność. „Robot autonomiczny” może mieć aspekty związane tylko ze stroną spektrum automatyzacji lub tylko ze strony autonomicznej, ale w rzeczywistości większość systemów robotów zaprogramowanych metodami sztucznej inteligencji ma mieszankę obu.

### Plany: generowanie kontra wykonanie

Generowanie i realizacja planu są ze sobą ściśle powiązane, ponieważ robot nie może wykonać planu, jeśli taki nie został wygenerowany. W zastosowaniach automatyki przemysłowej plan może być skonstruowany przez człowieka za pomocą pilota uczącego, aby pokazać robotowi, czego chce od niego aby zrobił dana osoba. Wysiłek obliczeniowy i inteligencja nie polega zatem na generowaniu planu, ale ogranicza się do zdolności robota do wykonywania lub wykonywania tych czynności dokładnie w ten sam sposób przy każdym przejściu i tak szybko, jak to możliwe. W przeciwieństwie do tego, aplikacje autonomiczne umożliwiają robotowi konstruowanie lub dostosowywanie samego planu, ponieważ różnice w misji lub środowisku, a nawet zużycie mechanicznych połączeń robota uniemożliwiają mu

wcześniejsze stworzenie kompletnego, w pełni sprawdzonego planu. Planowanie jest jednym z najbardziej aktywnych obszarów rozwoju sztucznej inteligencji, a społeczność ma sprawdzone wyniki w zakresie planowania misji, planowania awaryjnego, optymalnej alokacji zasobów i harmonogramowania oraz określania, kiedy należy używać wysoce zoptymalizowanych planów lub bardziej elastycznych planów uogólnionych. Podejścia kontrolne postrzegają realizację planu jako trudny lub interesujący problem; sztuczna inteligencja podchodzi do generowania widoku jako miejsca, w którym należy skoncentrować wysiłek.

### **Działania: deterministyczne kontra niedeterministyczne**

Spektrum działań deterministycznych i niedeterministycznych zasługuje na bardziej szczegółowe omówienie, biorąc pod uwagę, że badanie przeprowadzone w 2012 r. przez Defense Science Board, które wykazało, że ta dychotomia stanowi przeszkodę w tworzeniu skutecznych procedur testowania i oceny. Deterministyczny i niedeterministyczny to terminy używane w informatyce do rozróżniania typów algorytmów - zwykle stosowane do automatów skończonych (FSA). W deterministycznym AS, jeśli robot znajduje się w określonym stanie i otrzymuje określony zestaw danych wejściowych, możliwe jest tylko jedno wyjście. W algorytmie niedeterministycznym istnieje wiele możliwych wyników, a wybrany wynik zależy od innego czynnika lub zdarzenia. Brak determinizmu może wynikać z natury danych wejściowych: jeśli jest wiele danych wejściowych, dane wejściowe pochodzą z zaszumionych czujników lub zakres wartości dostarczanych przez czujniki jest bardzo duży, wówczas jawne wyrażenie danych wyjściowych dla wszystkich kombinacji wejść. Innym źródłem niedeterminizmu jest stan sprzętu. Jeśli robotowi wyczerpuje się moc, nie może odsunąć się tak daleko od przeszkody. Na przykład robot miał przesunąć się o 1m w prawo, ale przesunął się tylko o 0,75m. Jeśli robot ma wiele procesów działających jednocześnie, co jest prawdopodobne, wówczas czas między procesami wprowadza niedeterminizm. Na przykład algorytm przetwarzania czujnika w Próbie 1 może aktualizować bufor sekundę lub dwie po tym, jak algorytm sterowania silnikiem uzyskuje dostęp do bufora w celu sterowania następnym ruchem. W wersji próbnej 2 bufor może zostać zaktualizowany wcześniej. Tak więc w dwóch próbach robot wykonuje nieco inne ruchy, podobnie jak ludzie, którzy rzadko wchodzą w dokładnie to samo miejsce za każdym razem, gdy idą korytarzem. Ludzie mają wzorec chodzenia po korytarzu i można liczyć na to, że z powodzeniem przejdą korytarzem bez uderzania w innych ludzi lub ściany. Podobnie jest z niedeterminizmem: robot może zejść korytarzem i nie uderzyć człowieka, chociaż to, czy robot skręca w lewo, czy w prawo, może zależeć od tego, kiedy wykrył osobę, od wszelkich dźwięków czujnika, które mogą sprawić, że będzie myślał, że jest bliżej ściana niż była w tym momencie, a zatem powinna skręcać w lewo, a nie w prawo, i wszelkie inne bodźce zmysłowe lub synchronizacja wewnętrznych obliczeń. Powyższe przykłady sugerują, że systemy deterministyczne i niedeterministyczne powinny być testowane w inny sposób. Prowadzenie, nawigacja i sterowanie historycznie opierały się na algorytmach deterministycznych, prowadząc inżynierów do tworzenia eleganckich mechanizmów zapewniających poprawną, pełną kontrolę gwarantującą wygenerowanie określonego zestawu danych wyjściowych. Te tradycyjne metody są trudne do zastosowania w algorytmach sztucznej inteligencji, ponieważ deterministyczne wyrażanie systemu staje się trudniejsze, gdy robot staje się bardziej złożony i składa się z wielu działań działających równolegle. Chociaż teoretycznie niedeterministyczny automat skończony może zostać przekształcony w automat deterministyczny, w praktyce rozmiar i złożoność wejść, wyjść i względy czasowe sprawiają, że nie jest to realną opcją. Skala utrudnia projektantowi wizualizację lub testowanie wszystkich możliwych sytuacji, co utrudnia akceptację inteligentnych robotów. Autonomiczne możliwości robotyki AI były historycznie niedeterministyczne, a wydajność mierzono jako średnią lub prawdopodobieństwo statystyczne. Przykładem jest praca Pattie Maes i Roda Brooksa, w której sześciopięciopalcowy robot Czingis nauczył się chodzić. Nogi Czingisa muszą się poruszać w tym samym czasie, a niewielkie zmiany terenu lub sprzęt mogą zmienić manewrowość platformy. Zamiast próbować

zakodować wszystkie możliwe warunki, Czyngis obudził się z dwoma pragnieniami i trzynastoma zachowaniami. Miał ochotę iść do przodu, mierzoną przez wleczone koło, i nie leżeć na brzuchu, jak mierzone przez dwa czujniki dotykowe. Dwanaście prostych zachowań dotyczyło machania nogą do przodu lub do tyłu, a jedno do balansowania. Czyngis nauczył się chodzić średnio w mniej niż dwie minuty, próbując różnych ruchów nóg i otrzymując pozytywne i negatywne informacje zwrotne z czujników. Ale ogólne działania robota nie były deterministyczne, ponieważ nie wykonywał dokładnie takich samych ruchów nóg za każdym razem, gdy się budził, ponieważ warunki były nieco inne. Testowanie algorytmów niedeterministycznych jest otwartym pytaniem badawczym. Istniejące techniki testowania często polegają na statystycznym próbkowaniu przestrzeni danych wejściowych. Wielu użytkowników, zwłaszcza wojskowych, ma problem z brakiem determinizmu robota, ponieważ nie ma 100% pewności, że robot za każdym razem zrobi dokładnie to samo, mimo że ten niedeterminizm jest częścią tego, jak robot dostosowuje się do środowiska, obciążenie obliczeniowe lub stan sprzętu. Dyskomfort użytkownika jest często potęgowany, jeśli robot nie może pokazać, dlaczego robi to, co robi lub jeśli nie może osiągnąć swoich celów, co jest powszechnym błędem projektowym interakcji człowiek-robot omówionym w rozdziale 13. Nawet jeśli robot osiąga cel, to nie jest do końca przewidywalne, że sposób realizacji celu jest dla niektórych niepokojący. Filozofia robotyki AI brzmi: „jeśli chcesz, aby robot był tak inteligentny jak owad, pies lub człowiek, musisz zaakceptować, że robot będzie tak niedeterministyczny jak owad, pies lub człowiek”.

### **Modele: otwarty i zamknięty świat**

#### **MODEL ŚWIATA**

Aby system komputerowy mógł zrozumieć świat, musi istnieć odpowiednia obliczeniowo reprezentacja zwana modelem świata. Jeśli robot ma tylko możliwości nawigacyjne, model świata może składać się z mapy, które obszary są zajęte, a które puste. Przedstawienie świata niekoniecznie jest przestrzenne. Jeśli robot wchodzi w interakcję z innymi agentami, jak widać w robotyce społecznej, może być zmuszony do reprezentowania tego, co uważa za przekonania, pragnienia i intencje innych agentów. Model świata może być zbiorem modeli świata; Robot w fabryce chemicznej może mieć odwzorowanie podobne do mapy do rozumowania przestrzennego, model wszystkich rodzajów klamek zaworów, które byłyby w zakładzie chemicznym i sposób ich obracania, oraz model przekonań dla interakcji społecznych z innymi pracownikami. Model świata może być wstępnie zaprogramowany w robocie, może być wyuczony przez robota lub może być kombinacją obu. Wstępnie zaprogramowane części modelu świata to często mapy lub listy obiektów lub warunków, których może się spodziewać robot.

#### **ZAŁOŻENIE ŚWIATA ZAMKNIĘTEGO**

Modele świata są klasyfikowane jako świat zamknięty lub świat otwarty. Założenie zamkniętego świata mówi, że wszystko, co możliwe, jest znane a priori, że nie ma niespodzianek. W logikach formalnych oznacza to, że każdy obiekt, warunek lub zdarzenie, które nie jest określone w bazie danych, jest fałszywe. W bestsellerowej książce i filmie Jurassic Park inteligentny system monitorowania zaprojektowany do śledzenia dinozaurów zawiódł z powodu założenia o zamkniętym świecie. System nie zauważył wzrostu populacji dinozaurów, ponieważ założono, że dinozaury nie mogły się rozmnażać; w rezultacie system rozpoznawania obiektów alarmowałby tylko wtedy, gdy było mniej dinozaurów niż oczekiwano, i nigdy nie ostrzegał, gdyby było ich więcej.

#### **ZAŁOŻENIE OTWARTEGO ŚWIATA**

Algorytmy robota działającego w założeniu otwartego świata zakładają, że lista możliwych stanów, obiektów lub warunków nie może być do końca określona. Algorytmy te mogą wykorzystywać uczenie maszynowe do dodawania nowych obiektów lub zdarzeń do modelu. Jeśli świat jest reprezentowany

za pomocą logiki formalnej, stosuje się bardziej zaawansowaną logikę, która pozwala systemowi wycofać twierdzenia dotyczące świata, gdy obiekt jest poruszany lub gdy coś się zmienia; w przeciwnym razie nie można zagwarantować prawidłowego wnioskowania logicznego.

### **PROBLEM Z RAMKĄ**

Z założeniem zamkniętego świata wiąże się problem ramki. Problem ramki odnosi się do problemu poprawnej identyfikacji tego, co jest niezmiennie na świecie, a tym samym nie wymaga ciągłej aktualizacji, zmniejszając tym samym obliczenia. System, który wymaga przedstawienia i wnioskowania na temat wszystkich elementów lub stanów świata, prawdopodobnie nie będzie w stanie poradzić sobie z wynikającą z tego komplikacją obliczeniową. Rozważ na przykład próbę wypisania wszystkich elementów w pokoju (ściany, sufity, biurko, krzesło itp.). Lista bardzo szybko się wydłuża. Jednym z wkładów Shakeya w sztuczną inteligencję było to, że pomogło to w wykryciu problemu z ramkami. Robot poniósł znaczne koszty obliczeniowe, ponieważ został zaprogramowany do wyszukiwania i identyfikowania wszystkiego w pomieszczeniu przy każdej aktualizacji kamery. Innym aspektem problemu ramowego jest określenie, co jest istotne dla konkretnej zdolności. W Parku Jurajskim programista zastosował założenie zamkniętego świata, że dinozaury nie mogą się rozmnażać, aby zmniejszyć wymagane zasoby obliczeniowe; w przeciwnym razie program obserwacyjny musiałby szukać wszystkich dinozaurów na wszystkich obrazach. Czy w przypadku Shakeya, jeśli robot próbuje poruszać się po biurku, konieczne jest zidentyfikowanie obiektu blokującego jego drogę jako biurka lub krzesła? Z drugiej strony, jeśli robot szuka biurka, ważna jest identyfikacja biurka. Problem ramowy oznacza, że projektant robota musi być świadomy tego, w jaki sposób założenia dotyczące świata wpływają na adaptacyjność robota i jaka jest rola monitorowania ważności tych założeń. Dziedzina robotyki kognitywnej, mała grupa w robotyce AI, zajmuje się problemem ramowym, badając rozumowanie temporalne i logikę temporalną. Jednak ta książka koncentruje się na głównym nurcie robotyki behawioralnej, która stara się uniknąć problemu ramowego, minimalizując użycie modeli świata. Niebezpieczeństwa związane z założeniem zamkniętego świata i związany z nim problem z ramami wpływają zarówno na podejście inżynierskie, jak i AI do robotyki. Różnica między tymi dwoma podejściami polega na tym, że dobry system sterowania jest skrupulatnie zaprojektowany, aby uchwycić wszystko, co musi być w modelu światowym, podczas gdy system AI jest zaprojektowany tak, aby dostosować się do niedoskonałego modelu świata. W praktyce zarówno roboty z teorią sterowania, jak i AI mogą wykorzystywać model zamkniętego świata. Na szczęście model otwartego świata nie wymaga zaawansowanego uczenia maszynowego nowych obiektów lub stanów; jak zobaczymy, zwierzęta używają prostych modeli otwartego świata.

### **Reprezentacja wiedzy: symbole kontra sygnały**

Można dokonać rozróżnienia między zautomatyzowanymi i autonomicznymi zdolnościami w oparciu o formę informacji przetwarzanych przez robota: sygnały lub symbole. Automatyzacja oznacza, że robot reaguje na sygnały lub surowe dane, na przykład poruszaj się, aż enkoder zasygnalizuje osiągnięcie lokalizacji L1, aktywuj ruchy chwytania, przesuń się do lokalizacji L2, zwolnij. Przypominając siedem obszarów sztucznej inteligencji z Części 1, sztuczna inteligencja generalnie operuje na danych, które zostały przekształcone w informacje lub symbole, na przykład przejdź do dzbanka, chwyć, przejdź do MOJEJ FILIŻANKI i zignoruj SWÓJ FILIŻANKA. W robotyce często trudno jest robotowi rzetelnie rozpoznać obiekty lub zrozumieć sytuację, a zatem istnieje bariera w przekształcaniu nieprzetworzonych sygnałów czujnika w oznakowane obiekty lub obszary. Na szczęście w robotyce AI autonomiczne zachowania „niższego” poziomu opisane w Części 4 nie wykorzystują symboli i zamiast tego działają z sygnałami czujnika zwanymi afordancjami, podczas gdy zdolności deliberacyjne „wyższego” poziomu działają z symbolami.

## Ograniczona racjonalność

Autonomia może wywoływać strach lub nieufność ze strony użytkowników, pomimo zasady ograniczonej racjonalności. Nieufność często bierze się z dwóch źródeł: politycznej definicji autonomii i jej niedeterministycznej natury. Autonomia jest definiowana jako cecha lub stan bycia samorządnym, posiadania samodzielnej wolności, a zwłaszcza niezależności moralnej. W codziennym użyciu oznacza to, że ludzie mogą robić, co chcą, bez narzucanych politycznie granic. Wiele osób martwi się, że „autonomia” oznacza, że robot ma podobną samorealizowaną swobodę wyboru, a zatem będzie nieprzewidywalny lub wykaże inicjatywę wykraczającą poza zamierzenia projektanta. W zastosowaniu robotyki AI termin autonomia jest znacznie bardziej ograniczony. Użycie terminu „autonomia” wywodzi się z „autonomicznej” części definicji i wywodzi się z wynalezienia regulatora odśrodkowego kulowego w XVIII wieku. Regulator flyball wirowałby z ilością pary wytwarzanej przez silnik parowy. Im szybszy obrót, tym wyżej polecą zestawy piątek. Jeśli kule uniosłyby się na określoną wysokość, wytwarzało się zbyt dużo pary, co spowodowałoby wyłączenie systemu produkcyjnego, zapobiegając w ten sposób wybuchowi kotła. Regulator flyball nie jest praktycznie autonomiczny, tylko mechanicznie autonomiczny.

## OGRANICZONA RACJONALNOŚĆ

Jeden z założycieli AI, Herbert Simon, wykazał, że zdolność podejmowania decyzji przez wszystkich agentów, zarówno ludzkich, jak i sztucznych, jest ograniczona ilością posiadanych informacji, ich zdolnościami obliczeniowymi oraz ilością czasu dostępnego na podjęcie decyzji. Ta zasada ograniczonej racjonalności oznacza, że chociaż robot może dynamicznie dostosowywać się lub zmieniać plany, aby przewyższyć występowanie niemodelowanych zdarzeń w otwartym świecie, nie może wykraczać poza to, do czego został zaprogramowany. Zauważ, że chociaż Ghengis nauczył się nowej zdolności chodzenia, nie przekroczyła ona granic obliczeniowych; mógł nauczyć się tylko kombinacji swoich 13 początkowych zachowań. Ograniczona racjonalność i zrozumienie, że robot jest samorządny w sensie regulatora flyball, może nie pomóc ludziom zaufać robotom. Główną barierą w akceptacji zdolności autonomicznych jest pytanie Jak można ufać systemowi niedeterministycznemu? Biorąc pod uwagę trudności związane z wizualizacją i testowaniem niedeterministycznych algorytmów, projektanci lub użytkownicy mogą łatwo zaskoczyć się działaniami, jakie podejmuje robot, aby zmaksymalizować sukces w realizacji określonego celu lub misji.

## Wpływ automatyzacji i autonomii

W poprzedniej sekcji opisano, w jaki sposób sztuczna inteligencja i podejścia inżynierskie do inteligencji maszynowej reprezentują zasadniczo różne paradygmaty. Wybór paradygmatu przy projektowaniu systemu autonomicznego wpływa na styl programowania i projekt sprzętu dla robota oraz prawdopodobieństwo wystąpienia niektórych typów awarii funkcjonalnej i błędu ludzkiego. Zdolności autonomiczne mogą być tak prymitywne, że nie da się ich odróżnić od automatyzacji, ale różnice między nimi stają się coraz bardziej wyraźne wraz ze wzrostem liczby i zaawansowania możliwości. W ten sposób wpływ wyboru charakterystyki ze strony automatyzacji spektrum suwaków lub ze strony autonomii będzie w przyszłości bardziej wyraźny.

## Wpływ na styl programowania

Styl programowania autonomii, a nawet języki programowania mogą być różne. Jak zauważono w poprzedniej sekcji, zdolności autonomiczne są generalnie obliczeniowo niedeterministyczne. W robotyce AI ogólna reakcja robota wynika z interakcji niezależnych modułów, a nie jest wyraźnie określona. Jak zobaczymy w pozostałych częściach, podstawowa funkcjonalność sterowania robotem jest wyrażona w modułach zwanych zachowaniami. Zachowania są często programowane w C++ lub

innym języku proceduralnym. Bardziej zaawansowane funkcje deliberatywne są uchwycone w bardziej wyrafinowanych modułach, które mogą działać wolniej niż zachowania, a zatem są asynchroniczne z szybkim sterowaniem robotem w czasie rzeczywistym. Funkcje deliberatywne można zaprogramować w Lisp, funkcjonalnym języku programowania, który jest szczególnie odpowiedni do planowania i rozumowania.

### **Wpływ na projekt sprzętu**

Podczas projektowania sprzętu należy uwzględnić autonomiczne możliwości. Przypomnijmy, że autonomia wymaga wyczuwania w celu dostosowania się do kluczowych elementów dynamicznego świata. Projektowanie zazwyczaj skupiało się na umożliwieniu robotom działania, koncentrując się na efektorach, mocach i komunikacji bezprzewodowej, ale ignorując podsystemy sterowania i wykrywania. Jak zostanie podkreślone, począwszy od części 6, na inteligentne roboty należy patrzeć z perspektywy projektowania ekologicznego: Jakich możliwości potrzebuje robot, aby wykonywać swoje funkcje w określonym środowisku? Użytkownicy często wymagają, aby robot bezprzewodowy miał wbudowaną funkcję powrotu do domu. Jeśli komunikacja bezprzewodowa zostanie utracona, robot automatycznie wróci do pozycji wyjściowej. Jednak wiele robotów naziemnych zaprojektowano bez wbudowanego komputera; wszystkie obliczenia są wykonywane w jednostce sterującej operatora. Bez wbudowanego komputera nie ma możliwości dodania funkcji powrotu do domu. Jednak brak komputera na pokładzie zmniejsza pobór mocy robota, pomaga zmniejszyć wagę i rozmiar robota, zmniejsza koszty i zwiększa niezawodność robota, biorąc pod uwagę, że komputer może nie przetrwać w trudnych warunkach. Późniejsze dodanie nowych czujników do nieinteligentnego robota może być niemożliwe ze względu na zmianę rozmiaru i wagi robota, brak pojemności na wewnętrznych szynach elektrycznych i zasilających, a nawet brak powierzchni do montażu. W wielu typach załogowych i bezałogowych statków powietrznych człowiek nie może utrzymać stabilnego lotu, podczas gdy komputer może reagować szybciej i wykonywać nieintuicyjne, ale krytyczne ruchy wielu powierzchni sterowych. Ale te czujniki wymagane do stabilności lotu nie są wystarczające do autonomicznych zdolności, takich jak unikanie przeszkód. Ostatnio, ze względu na trudności z modernizacją czujników w UAV, podjęto próby ponownego wykorzystania kamery w robocie do takich funkcji, jak unikanie przeszkód, wykorzystanie obszaru sztucznej inteligencji w zakresie widzenia komputerowego. Jeśli człowiek ma trudności w kierowaniu robotem z powodu braku wykrywania, ani automatyzacja, ani autonomia nie są w stanie zrobić tego lepiej. Inuktun Mine Cavern Crawler, używany do poszukiwań i ratownictwa w kanionie Crandall w stanie Utah w 2007 roku, był trudny do kontrolowania przez człowieka i niemożliwe było dodanie do niego jakichkolwiek autonomicznych zdolności. Platforma robota składała się z dwóch głównych elementów: śledzonej podstawy robota z kamerą z funkcją obrotu i pochylecia i zbliżenia oraz modułu świateł. Moduł kamery został zapakowany w cylindryczny maszt, który leżałby płasko za podstawą, podczas gdy robot był opuszczany przez otwór, a następnie podnoszony o 90 stopni nad podstawą robota. Robot powstał w trzy dni z zamiarem stworzenia możliwie najmniejszego, wodoodpornego robota. W wyniku decyzji projektowych nie było czujnika ani mechanicznego ogranicznika podnoszenia masztu ze względu na jego rozmiar i dodatkową złożoność elektroniczną. W konsekwencji, jeśli maszt został podniesiony powyżej 90 stopni, istniała możliwość, że utknie w tej pozycji i nie będzie w stanie złożyć się z powrotem i pozwolić robotowi na ponowne wejście do odwiertu. Operatorzy robotów zostali poinstruowani, aby liczyć podczas podnoszenia masztu i nigdy nie podnosić masztu na czas dłuższy niż 10, zakładając, że maszt był zawsze podnoszony z neutralnej pozycji płaskiej. Jednak robot często musiałby zmieniać kąt masztu, aby mógł poruszać się w bardzo ograniczonych przestrzeniach zawałenia, co wymagało od operatorów mentalnego utrzymywania szacunkowej pozycji masztu. Bez enkoderów na maszcie nie było niezawodnego i dokładnego sposobu na zautomatyzowanie wznoszenia, aby zatrzymać się przed 90 stopniami od dowolnej pozycji startowej lub stworzenie systemu ostrzegawczego.

## **Wpływ na rodzaje awarii funkcjonalnych**

Automatyzacja i autonomia mają podobne tryby awarii, ale różnią się rodzajem rozwiązań. Dwoma głównymi rodzajami awarii inteligencji robotów są awarie funkcjonalne, w których robot robi to, do czego został zaprogramowany, ale nie to, co było zamierzone, oraz awarie wynikające z mitu zastępowania, że maszyna może zastąpić człowieka bez zmiany cegokolwiek innego w systemie, takiego jak wyświetlacze, procedury i szkolenia. Teoretycznie system o zdolnościach autonomicznych powinien mieć mniej błędów funkcjonalnych niż system zautomatyzowany działający zgodnie z restrykcyjnym zestawem założeń świata zamkniętego. Ale w praktyce robot z autonomicznymi zdolnościami mogą nie mieć zdolności rozwiązywania problemów i rozumowania, które umożliwiłyby wyjście poza de facto zamknięty świat. Inżynierowie kognitywni i przemysłowi, którzy badają systemy człowiek-maszyna, mają długą listę przyczyn awarii systemów. Wiele z nich jest przypisywanych błędom ludzkim przez użytkownika maszyny lub robota, ale w rzeczywistości są to błędy ludzkie zaprojektowane przez projektantów. Niedawne badanie trybów awarii w robotyce katastrof wykazało, że około 50% awarii przypisywano błędom ludzkim, chociaż zauważono, że nie jest jasne, co człowiek mógł zrobić inaczej. Można dodać autonomiczne funkcje, aby poprawić wydajność robota w tych obszarach lub inteligentnie pomóc użytkownikowi.

## **Awarie funkcjonalne**

Jeśli oczekiwania robota dotyczące otoczenia są nieprawidłowe w złożonym świecie, może uzyskać odpowiednik „widzenia tunelowego” i przeoczyć rzeczy. Jak wspomniano wcześniej, w automatyzacji manipulatorów przemysłowych reprezentacja świata może obejmować założenie, że przedmiot, który ma być manipulowany, spawany lub malowany, znajduje się w precyzyjnym miejscu i orientacji. Jeśli to założenie zostanie naruszone, np. gdy taśmociąg zostanie zatrzymany, robot będzie manipulował, spawał lub malował cienkie powietrze. Podobne zjawisko może mieć miejsce w bardziej autonomicznych systemach. Wczesny autonomiczny system przetwarzania obrazu wojskowego USA uznano za poważną awarię podczas rzeczywistego starcia, ponieważ nie był w stanie zidentyfikować ruchów wojsk wroga. Po dalszej analizie odkryto, że moduły wykrywania obiektów i rozumienia sceny prawidłowo identyfikowały ruchy wojsk. Jednak algorytmy modułu zazwyczaj generowały fałszywe alarmy, więc system przekazał dane wyjściowe do innego modułu, który uzasadniał obrazy wykorzystując swoją wiedzę o najlepszych praktykach operacji wojskowych. Na nieszczęście dla systemu sztucznej inteligencji w tym konkretnym starciu oddziały wroga szybko straciły dyscyplinę i nie walczyły według zasad doktryny wojskowej. Tak więc, podczas gdy moduły wykrywania obiektów i rozumienia sceny dokładnie identyfikowały ruchy wojsk, inny moduł pomijał poprawną identyfikację z powodu sformalizowanego oczekiwania, że żołnierze nigdy nie będą walczyć w nieorganizowany sposób. Wracając do dyskusji na temat światów zamkniętych i otwartych, zarówno manipulator przemysłowy, jak i agent oprogramowania działały w świecie zamkniętym, ale system przetwarzania obrazu działał w świecie otwartym. Decyzja projektowa, by sformalizować oczekiwania dotyczące ruchów wojsk bez dodawania jakichkolwiek mechanizmów, aby zauważyć (lub pokazać człowiekowi), że oba moduły w zaskakujący sposób nie zgadzały się ze sobą, narzuciła de facto założenie zamkniętego świata na to, co powinno być systemem otwartego świata. W wyniku narzucenia projektu zmniejszono ogólną inteligencję agenta. Sztuczna inteligencja dąży do systemów, które mogą działać w otwartym świecie, ale implementacje mogą nie spełniać tych wygórowanych oczekiwań.

## **Wpływ na rodzaje błędów ludzkich**

### **MIT SUBSTYTUCJI**

Chociaż automatyzacja i autonomia mają na celu wyeliminowanie lub zmniejszenie zaangażowania człowieka, te możliwości często wprowadzają błąd człowieka lub operatora, ponieważ każdy robot jest



częścią większego systemu człowiek-maszyna. „Autonomiczny” często oznacza dla użytkownika, że robot jest bardziej zdolny niż jest w rzeczywistości, szczególnie w subtelny sposób, który sprawia, że ludzie są tak elastyczni. Założenie to określane jest mianem mitu substytucji w inżynierii kognitywnej<sup>220</sup>: mitu, że maszyna doskonale zastąpi człowieka w tym konkretnym zadaniu.

### **PROBLEM KONTROLI CZŁOWIEKA POZA PĘTLI (OOTL)**

Z szerszej perspektywy człowiek-maszyna, bez względu na to, jak niezależny od człowieka jest robot, jeśli coś pójdzie nie tak, oczekuje się, że człowiek przejmie rolę robota lub wstawi się za robotem. Ale może nie być kognitywnie możliwe, aby człowiek zareagował wystarczająco szybko lub zrozumieć, co robił robot i co jest nie tak, oraz określić, dlaczego, a następnie poprawnie rozwiązać problem. Zjawisko to jest dobrze znane w inżynierii kognitywnej od lat 80. XX wieku i jest ogólnie określane jako problem kontroli człowieka poza pętlą (OOTL). Problem ludzkiego OOTL okazał się śmiertelny we wczesnym autopilocie. systemy, w których piloci komercyjnych linii lotniczych wykonywaliby papierkową robotę lub inne obowiązki, gdy autopilot nagle wyłączałby się z powodu wystąpienia sytuacji, na którą nie był zaprogramowany. Piloci mogli mieć tylko kilka minut na ustalenie, co skłoniło autopilota do wyłączenia; co gorsza, ponieważ autopilot działał dobrze w rutynowych sytuacjach, z definicji powinien sam się wyłączyć tylko w przypadku wystąpienia nienormalnego zdarzenia. Fakt, że autopilot się wyłączył, oznaczał, że sytuacja będzie trudna do zdiagnozowania. Piloci często byli przyłapywani na nieświadomości i nie mogli przejść z tego, co aktualnie robili, z powrotem do pełnego zrozumienia tego, co się dzieje, co prowadziło do awarii. Zjawisko OOTL jest szczególnie szkodliwe w małych bezzałogowych statkach powietrznych, które latają na niższych wysokościach, gdzie operator może mieć tylko sekundy na zdiagnozowanie i odzyskanie systemu. Ogólnie rzecz biorąc, człowiek będzie w stanie przezwyciężyć problem sterowania OOTL, jeśli system jest zaprojektowany tak, aby umożliwić człowiekowi łatwą wizualizację wewnętrznego stanu robota i ułatwić płynne przekazanie sterowania. W przypadku lotnictwa załogowego autopiloty zaczęły wskazywać, kiedy samolot zbliżał się do granic swojego reżimu kontroli, ostrzegając pilotów, aby zaczęli koncentrować się na samolocie i przygotowywać się do przejęcia. Badania człowiek-maszyna wykazały, że problem OOTL występuje również w automatyzacji fabryk i że mniej automatyzacji lub autonomii może być bardziej opłacalne. Endsley i Kaber odkryli, że w przypadku wysoce zautomatyzowanych fabryk osoba, która nie była bezpośrednio związana z operatorami (być może z sąsiedniego biura wykonującego papierkową robotę), potrzebowała więcej czasu na zdiagnozowanie i naprawienie problemu, który spowodował wyłączenie linii produkcyjnej. Zmniejszono czas i utratę dochodów, jeśli dana osoba otrzymała pracę, która wymagała zaangażowania się w automatyzację; dzięki zaangażowaniu dana osoba miała wyższy poziom świadomości sytuacyjnej tego, co robi system, czy wpada w kłopoty i jakie mogą być alternatywy w przypadku pojawienia się problemu. Wniosek był taki, że taniej było zapłacić, aby pracownik był zatrudniony w pełnym wymiarze godzin, aby rozwiązywać rzadkie, ale trudne problemy, niż eliminować pracownika na pełny etat i wzywać go tylko wtedy, gdy pojawił się problem. Nie brano pod uwagę trzeciej alternatywy polegającej na dodaniu możliwości sztucznej inteligencji do rozwiązywania problemów i wnioskowania. Wracając do mitu substytucji, chodzi o to, że chociaż zarówno automatyzacja, jak i autonomia zapewniają możliwości, które mogą wydawać się zastępować ludzkie możliwości, inteligencja maszyn różni się od inteligencji ludzkiej. Projektanci muszą włożyć dodatkowy wysiłek, aby upewnić się, że maszyna ma dorozumianą inteligencję, którą wykazywałby człowiek w określonej sytuacji, tak aby robot rzeczywiście zastępował kluczowe atrybuty człowieka lub aby granice miejsca, w którym robot nie zachowywał się jak człowiek jest zrozumiany. Projektanci muszą również zdawać sobie sprawę z tego, że stosunkowo niewiele wiadomo na temat interakcji między inteligencją człowieka a inteligencją maszyny, że wystąpią awarie, a zatem mechanizmy umożliwiające człowiekowi przejęcie kontroli są niezbędne w praktyce.

## **Przestrzenie robocze w dodawaniu autonomicznych możliwości**

Kwestia projektowania zdolności autonomicznych została przeanalizowana przez Departament Obrony USA. Badanie przeprowadzone w 2012 r. przez US Defense Science Board (DSB) wykazało, że systemy bezzałogowe, niezależnie od tego, czy są używane na ziemi, w powietrzu czy na morzu, mają problemy wynikające z początkowych decyzji projektowych. Badanie wykazało, że systemy były często programowane z możliwościami, które są zbyt wąskie, aby były naprawdę przydatne, mają przekroczone koszty oprogramowania i sprzętu do przeprojektowania oraz zwiększają, zamiast zmniejszać, liczbę osób potrzebnych do wykonania misji. Podczas gdy systemy bezzałogowe ograniczyły liczbę zagrożonych osób, do obsługi platform fizycznych oraz nadzorowania operacji i przejmowania kontroli w przypadku awarii oprogramowania potrzebna jest duża liczba osób. Ponadto badanie wykazało, że głównym problemem było to, że możliwości autonomiczne nie były wykorzystywane w odpowiednich aplikacjach lub nie były wdrażane zgodnie z najlepszymi praktykami ze sztucznej inteligencji i kognitywistyki. Niedopasowanie aplikacji i niezgrabne wdrożenia wywoływały nieufność wśród użytkowników.

## **PRZESTRZEŃ ROBOCZA W SYSTEMACH CZŁOWIEK-MASZYNA**

Badanie zawierało liczne rekomendacje; po pierwsze, projektanci i urzędnicy ds. zakupów rozważają projekt systemu bezzałogowego pod kątem pięciu przestrzeni handlowych w zakresie sprawności systemów człowiek-maszyna, planów, wpływu, perspektywy i odpowiedzialności. Poniżej opisano pięć pól handlowych:

\* Sprawność fizyczna, która reprezentuje kompromis między optymalnością a odpornością. Manipulatory przemysłowe są zaprojektowane tak, aby wykorzystywać wewnętrzne czujniki położenia do wykonywania optymalnych ścieżek w zestawie oczekiwanych działań. Taka konstrukcja umożliwiła zoptymalizowanie robota pod kątem szybkiego wykonania i obniżenie kosztów, ponieważ nie wymagał żadnych zewnętrznych czujników. Aby osiągnąć tę optymalność, większość projektantów założyła, że manipulowany obiekt znajduje się we właściwym miejscu i nic nie znajduje się w przestrzeni roboczej robota. W rezultacie powstał robot, który został zaprojektowany dla konkretnej linii technologicznej, ale trudno go było przystosować do nowego zestawu obiektów, którymi trzeba manipulować lub na których działać, i mógł zabić pracownika, który wszedł do przestrzeni roboczej.

\* Plany, które stanowią kompromis między wydajnością a dokładnością. Przykładem tego kompromisu był omówiony wcześniej algorytm, który został użyty do liczenia dinozaurów w Parku Jurajskim. Było to bardzo wydajne, ale kosztem dokładności. Bardziej inteligentne systemy wykorzystują postęp w metodach AI do zmniejszania zależności od założeń i monitorowania realizacji planów. W nawigacji jednym podejściem jest planowanie pełnej ścieżki całej misji z wysokim stopniem rozdzielczości przestrzennej, co jest dokładne, podczas gdy innym podejściem jest planowanie z niskim stopniem rozdzielczości przestrzennej i reagowanie na napotykaną przeszkodę, co jest wydajny.

\* Wpływ, który reprezentuje kompromis między perspektywą scentralizowaną a rozproszoną. Jak zostanie opisane w dalszej części, zespoły robotów mogą być sterowane za pomocą scentralizowanego komputera lub przy kontroli rozproszonej między zespół. Scentralizowana kontrola wprowadza opóźnienia w komunikacji i luki w zabezpieczeniach w przypadku awarii centralnego komputera. Badacze sztucznej inteligencji wykazali skuteczność kontroli rozproszonej, gdy roboty koordynują między sobą, używając prostych zachowań, aby działać jako rój lub licytować bardziej złożone zadania.

\* Perspektywy, które reprezentują kompromis między poglądami lokalnymi i globalnymi. Częstym dylematem w nawigacji pojazdów naziemnych jest problem kanionu skrzynkowego, w którym robot zamienia się w ślepy zaułek. Jeśli robot jednocześnie nie mapuje świata i nie nadąża za swoją pozycją

w świecie, czyli tworzy globalny pogląd, robot może nie zauważyć, że znajduje się w ślepych zaułku. Robot działający tylko w widoku lokalnym może być podatny na efekt „latania w oknie”. Muchy są przyciągane przez światło, a gdy znajdują się w pokoju lub samochodzie, podchodzą do okna. Ale nie są w stanie zobaczyć szyby i nadal próbują przez nią przelecieć, często tracąc otwarte okno w innym miejscu. Jednak tworzenie i utrzymywanie dokładnej globalnej mapy świata jest kosztowne obliczeniowo, wymaga wykrywania zasięgu w celu modelowania przestrzennego i zrozumienia sceny.

\* Odpowiedzialność, która reprezentuje kompromis między celami krótkoterminowymi i długoterminowymi oraz decyzję o tym, który agent jest ostatecznie odpowiedzialny za osiągnięcie tych celów. Zdolności autonomiczne mogą dotyczyć określonej fazy misji. Na przykład niektóre statki powietrzne mają zdolność autonomicznego startu, ale nie mają możliwości autonomicznego lądowania. W prawie każdym przypadku oczekuje się, że człowiek przejmie kontrolę, jeśli autonomia nie może być wykorzystana lub zawiedzie, więc człowiek ponosi odpowiedzialność nawet za autonomiczny start lub lądowanie. W przeciwieństwie do tego, sonda NASA Deep Space 1 z 1998 roku została zaprojektowana tak, aby była w pełni autonomiczna dla całej misji z pokładowym rozwiązywaniem problemów i możliwościami wnioskowania w celu obejścia awarii na pokładzie. (Inżynierowie dokonali rekonfiguracji systemu nawigacyjnego podczas pierwszego uruchomienia próbnego, ale później sonda robota była już sama).

Wracając do dyskusji w poprzedniej części dotyczącej różnic między automatyzacją a autonomią, łatwo zauważyć, w jaki sposób mogą pojawić się problemy projektowe i jak testowanie deterministyczne może je pominać. Przestrzenie handlowe fitness i planu ilustrują napięcia między założeniami otwartego i zamkniętego świata, w których algorytmy i działania mogą być bardzo precyzyjne w dobrze rozumianych sytuacjach, ale nieodpowiednie do wszystkiego, co wykracza poza oczekiwania. Twierdzenia o optymalności i dokładności opierają się na pewnej ograniczonej reprezentacji świata, co oznacza, że system może być podatny na zbyt restrykcyjne założenie o zamkniętym świecie i słabą obsługę problemu ramki. Bez odpowiedniej równowagi sprawności, planowania, perspektywy i obowiązków system może utknąć w realizacji niewłaściwego planu lub mieć trudności z jego korektą. Jednak człowiek może nie zauważyć lub nie mieć dostępu do szczegółowych prac potrzebnych do modyfikacji planu. Jak zostanie omówione później w Interakcjach między człowiekiem a robotem, te przestrzenie handlowe uwzględniają również potencjalne przyczyny problemów między operatorami a robotami. Na przykład przestrzenie wpływu i perspektyw obejmują projektowanie interfejsów, w których interfejs potrzebny jednej osobie do kierowania i przeglądania wszystkich informacji gromadzonych przez robota może bardzo różnić się od interfejsu, który umożliwia wielu osobom w różnych lokalizacjach korzystanie z informacji istotne dla nich. Dopasowanie, wpływ, perspektywy i odpowiedzialność przestrzeni handlowe są szczególnie podatne na mit substytucji i wprowadzają zagadki kontroli poza pętlą. Kompromisy między centralizacją a systemami rozproszonymi obejmują również koordynację między ludźmi korzystającymi z robota, gdzie jeden zespół operatorów może być odpowiedzialny za interpretację obrazów, a inny za działanie na obrazach. Jeśli człowiek pracuje w widoku lokalnym, ale otrzymuje wszystkie dane do widoku globalnego, człowiek może być przeciążony danymi i podejmować decyzje dłużej. Jeśli chodzi o przestrzeń handlową odpowiedzialności, sukces z celami krótkoterminowymi buduje zaufanie do systemu, które może przewyciężyć ogólną niechęć do delegowania celów długoterminowych na robota.

## **Streszczenie**

Ten rozdział odpowiada na cztery często zadawane pytania dotyczące automatyzacji i autonomii. Podstawowe pytanie brzmi: Jaka jest różnica między automatyzacją a autonomią? Jednym ze sposobów opisanie różnicy jest stwierdzenie, że automatyzacja dotyczy robotów jako narzędzi, a autonomia dotyczy robotów jako agentów, zgodnie z definicjami z Części 2. Innym sposobem jest

omówienie różnic pod kątem czterech głównych cech. Jedną z cech są plany: czy system koncentruje się na realizacji planów, czy na generowaniu planów? Kolejną cechą charakterystyczną są działania: czy system wykorzystuje algorytmy deterministyczne czy niedeterministyczne? Typ modelu świata pomaga również odróżnić narzędzia od agentów. Narzędzia zazwyczaj działają w zamkniętym świecie z dużą ilością informacji a priori, podczas gdy agenci pracują przy założeniu otwartego świata, aby wykonywać zmienne zadania w dynamicznie zmieniających się środowiskach. Czwarta różnica to reprezentacja wiedzy, niezależnie od tego, czy system manipuluje sygnałami, czy symbolami. W praktyce inteligentne roboty mają mieszankę cech, na przykład jeden robot może mieć więcej cech z zakresu automatyzacji, podczas gdy inny robot miałby więcej cech z zakresu autonomii. Jeśli inteligentne systemy są mieszanką atrybutów automatyzacji i autonomii, dlaczego to ważne, że istnieje różnica między autonomią a automatyzacją? Automatyzacja i autonomia są często nie do odróżnienia w przypadku stosunkowo mało inteligentnych zadań lub mogą być używane razem, na przykład nawigowanie do punktu nawigacyjnego, w którym algorytm sztucznej inteligencji może być używany do wybierania punktów nawigacyjnych dla bezzałogowego statku powietrznego i teorii sterowania, aby utrzymać kurs UAV. Jednak te podejścia odzwierciedlają różne paradygmaty projektowania, które wymagają różnych nakładów pracy i wprowadzą różne kompromisy w celu uzyskania bardziej inteligentnych możliwości. Wybór paradygmatu jest podobny do wybierania układów współrzędnych w rachunku różniczkowym, gdzie problem można rozwiązać, umieszczając rozwiązanie w układzie współrzędnych euklidesowych lub sferycznych. Jeśli zrobisz to poprawnie, obie dadzą tę samą odpowiedź, ale jeden będzie wymagał znacznie innego wysiłku i zwiększy prawdopodobieństwo błędu matematycznego. W robotyce wybór podejścia ma wpływ na styl programowania, projekt sprzętu, rodzaje błędów funkcjonalnych i rodzaje błędów ludzkich, na które należy się przygotować. Dlatego ważne jest, aby projektant świadomie wybrał właściwy koniec spektrum automatyzacji/autonomii, zapewniając określoną inteligentną zdolność i broniąc się przed różnymi rodzajami luk związanych z wybranym podejściem. Jakie są zalety autonomii nad automatyzacją? Czy możesz mi powiedzieć, kiedy użyć jednego, a kiedy drugiego? Ile potrzebuję autonomii? Zalety i przydatność funkcji autonomicznych w porównaniu z funkcjami zautomatyzowanymi zależą od aplikacji. Brak zasad projektowych określających, kiedy należy korzystać z autonomii, potęguje brak niedeterministycznych metod testowania; dobry osąd jest potrzebny, aby zdecydować, kiedy i jak dodać autonomię, a tradycyjne sposoby testowania nie gwarantują wykrycia jakichkolwiek błędów w ocenie. Chociaż nie ma sztywnych i szybkich zasad dotyczących tego, które możliwości najlepiej sprawdzają się w przypadku danego systemu robotycznego, projektant musi wziąć pod uwagę co najmniej pięć obszarów handlowych: sprawność, plany, wpływ, perspektywę i odpowiedzialność. Klasyczna automatyzacja w robotach produkcyjnych optymalizuje przydatność fizycznego robota pod względem precyzji, powtarzalności i kosztów, ale kosztem wymagania idealnego rozmieszczenia części dla robotów i ryzyka przystosowania robota do nowych zadań. Dodanie autonomicznej zdolności, takiej jak planowanie ścieżki, niezależnej od innych powiązanych funkcji autonomicznych, takich jak monitorowanie realizacji misji, w celu upewnienia się, że zaplanowana ścieżka jest nadal poprawna, grozi awariami systemu lub odrzuceniem przez użytkowników, którzy uważają system za zbyt trudny lub wymagający posługiwać się. Projektanci mogą nie docenić zakresu autonomii potrzebnej inteligentnemu robotowi do spełnienia ukrytych oczekiwań. Wynika to po części z mitu substytucji, że maszyna lub system oprogramowania może bezpośrednio zastąpić zdolność zapewnianą przez osobę. Zwykle osoba zapewnia zarówno rzeczywistą zdolność, taką jak planowanie, dokąd się udać, jak i dodatkowe dorozumiane możliwości, takie jak obsługa wyjątków lub rozwiązywanie problemów. Rozważ przykład z planów przestrzeni handlowej. Smartfony i urządzenia mobilne mogą obecnie generować optymalne trasy na podstawie map i, w niektórych przypadkach, informacji o ruchu drogowym. Jednak polegają na tym, że ludzki kierowca zauważy, że mapa lub trasa zawiera błąd lub, w przypadku operacji wojskowej, że blokada na drodze jest dziwna i może sygnalizować zasadzkę.

Dodawanie możliwości planowania trasy bezzałogowego pojazdu naziemnego bez dodawania możliwości monitorowania realizacji planu może stworzyć robota, który jest zbyt wąsko „autonomiczny”, aby mógł być używany bez stałego nadzoru człowieka lub bez ryzyka awarii. Choć sztuczna inteligencja ma algorytmy dla większości inteligentnych możliwości, to nadal od projektanta zależy określenie, który zestaw możliwości ma zostać uwzględniony w systemie.