ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1992
Seria: AUTOMATYKA z. 110 Nr kol. 1176

Jerzy Cyklis, Robert Czuła
Politechnika Krakowska

PRZYGOTOWANIE DANYCH DO SYMULACJI DUŻYCH ELASTYCZNYCH SYSTEMÓW PRODUKCYJNYCH PRZY UŻYCIU MODELU MACIERZOWEGO

SETTING-UP INPUT DATA FOR SIMULATION OF BIG FLEXIBLE MANUFACTURING SYSTEMS USING MATRIX MODEL

DATENVORBEREITUNG FÜR SIMULATION VON GROßEN FLEXIBLEN FERTIGUNGSSYSTEME BEI DER VERWENDUNG VON MATRIXMODELL

Streszczenie: W Instytucie Technologii Maszyn i Automatyzacji Produkcji Politechniki Krakowskiej opracowano model macierzowy symulacji elastycznych systemów produkcyjnych (ESP). Algorytm symulacji oparty jest na obliczaniu macierzy stanu oraz czasu bieżącego na każdym etapie symulacji. W pracy przedstawiono zmodyfikowaną metodę przygotowania danych dla programu komputerowego pracującego w oparciu o powyższą metodę. Zaletą tej metody jest łatwość modelowania dużych i złożonych systemów, w których funkcje niektórych elementów są identyczne (np. alternatywne obrabiarki, przedmioty o identycznych marszrutach technologicznych).

Summary: The matrix model of FMS has been invented at the Cracow University of Technology, Production Engineering Institut. The algorithm of simulation consists in calculation the state matrix and current time at each stage i of the change of the system. In the paper a modified method of preparing input data for the simulation program is described. This method makes it easy to model large-scale and complex systems with identical functions of some objects (e.g. alternative tools, objects with identical technological routes).

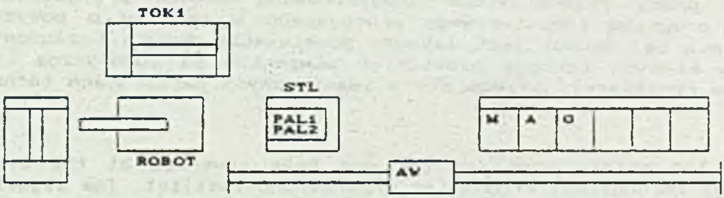
Zusammenfassung: In dem Institut für Maschinentechnologie und Fertigungsautomatisierung der Technischen Hochschule in Kraków ist ein Matrixmodell Simulation von flexiblen Fertigungssysteme (FFS) bearbeitet worden. Im Referat wurde eine modifizierte Methode der Datenvorbereitung für das Rechnerprogramm, das den Matrixmodell ausnützt, dargestellt. Der Vorteil der vorgeschlagenen Methode ist ihre Fähigkeit für Modellierung von großen, komplizierten Systemen, in den die Funktionen von manchen Objekten dieselbe sind (z.B. die alternativen Werkzeugmaschinen, die Werkstücke mit denselben technologischen Wegen).

1. Wstęp

Praca niniejsza jest rozwinięciem metody przygotowania danych do programu macierzowej symulacji elastycznych systemów produkcyjnych (ESP), zaproponowanej we wcześniejszych publikacjach [1,2,3]. Zaletą zmodyfikowanej metody jest łatwość modelowania dużych i złożonych systemów, w których funkcje niektórych elementów są identyczne (np. alternatywne obrabiarki, przedmioty o identycznych marszrutach technologicznych). Prowadzi to do występowania w zbiorze danych powtarzających się sekwencji analogicznych czynności. Algorytmiczne powielanie raz zapisanej sekwencji czynności wraz ze zbiorami wyjść umożliwi znaczne zmniejszenie pracochłonności przygotowania danych.

2. Model macierzowy

Dokładny opis macierzowej metody symulacji elastycznych systemów produkcyjnych zawarto w pracach [1,2,3]. Obecnie przedstawione zostaną jedynie podstawowe założenia dotyczące modelu macierzowego. Prezentacja ta jest konieczna dla zrozumienia przedstawionego w dalszej części referatu algorytmu przygotowania danych dla modelowania dużych systemów produkcyjnych. Na rysunku 1 przedstawiono uproszczoną wersję elastycznego systemu składającego się z : dwóch obrabiarek (TOK1, TOK2), stołu (STL), robota (ROBOT), magazynu przedmiotowego (MAG) oraz automatycznego wózka (AW). Do transportu dwóch rodzajów obrabianych przedmiotów (PO1, PO2) stosuje się odpowiednio palety PAL1 i PAL2.



Rys.1. Uproszczony model ESP
Fig.1. Simplified model of FMS

Automatyczny wózek pobiera z magazynu paletę pierwszego lub drugiego rodzaju, załadowaną przedmiotami przygotowanymi do obróbki i dostarcza ją na stół obrabiarek. Stąd poszczególne przedmioty transportowane są przy użyciu robota na jedną z dwóch pracujących obrabiarek. Po obrobieniu przedmioty (PO1 lub PO2) są ponownie ładowane na paletę. Po zakończeniu obróbki wszystkich przedmiotów znajdujących się na palecie automatyczny wózek odwozi je do magazynu.

Części systemu, z których każda jest traktowana jako funkcjonalna całość (np. obrabiarki, palety, wózki), tworzą zbiór elementów (k = 1, ..., K). Czynności wykonywane przez elementy systemu są zdyskretyzowane, tworząc zbiór arbitralnie wyodrębnionych tzw. czynności elementarnych (j = 1, ... J). Podział ten jest zależny jedynie od wymaganej dokładności modelu.

Dla uproszczenia i zapewnienia możliwości komputerowej obróbki danych przyjęto jednolity i uproszczony format zapisu nazw czynności elementarnych (Tablica I).

Pierwsza nazwa przed przecinkiem odnosi się do elementu przekazywanego lub podlegającego przemianom. Następne nazwy wskazują skąd i dokąd element jest przekazywany (przypadek strzałki "-->") lub gdzie podlega przemianom (przypadek gwiazdki "*").

PO.PAL --> TOK oznacza, że przedmiot obrabiany PO zostaje pobrany z palety PAL i załadowany na tokarkę TOK

PO.TOK* oznacza, że przedmiot obrabiany PO podlega obróbce na tokarce TOK

Tablica I

Zbiór elementów i czynności elementarnych

Elementy	Czynności	
	START	
MAO	PAL1.MAO-->AW	PAL2.MAO-->AW
AW	PAL1.AW-->STL	PAL2.AW-->STL
TOK1	PO1.PAL1-->TOK1	PO2.PAL2-->TOK1
TOK2	PO1.PAL1-->TOK2	PO2.PAL2-->TOK2
STL	PO1.TOK1*	PO2.TOK2*
PAL1	PO1.TOK2*	PO2.TOK2*
PAL2	PO1.TOK1-->PAL1	PO2.TOK1-->PAL2
PO1	PO1.TOK2-->PAL1	PO2.TOK2-->PAL2
PO2	PAL1.STL-->AW	PAL2.STL-->AW
	PAL1.AW-->MAO	PAL2.AW-->MAO
		END

Przedstawione powyżej zasady tworzenia nazw czynności mają tylko dwa wyjątki; są to czynności START i END. Fikcyjna czynność START ma za zadanie wprowadzenie warunków początkowych pracy systemu, natomiast wykonanie czynności END sygnalizuje zakończenie symulacji.

Dla każdej czynności j i uczestniczących w niej elementów k definiowane są tak zwane zbiory wyjść Out(j,k). Out(j,k) jest to zbiór czynności, do których wykonania przygotowany jest element k po zakończeniu czynności j. Tablica II przedstawia zbiory wyjść dla systemu prezentowanego na rys.1.

Tablica II

Zbiory wyjść dla modelowanego systemu

Element		MAO	AW	TOK1	TOK2	STL	PAL 1	PAL 2	PO1	PO2	RO-BOT
Czynność	k / j	1	2	3	4	5	6	7	8	9	10
START	1	2,11, 12,21	2,10, 12,20	4,14	5,15	3,13	2	12	2	12	#
PAL1.MAO-->AW	2	2,11, 12,21	3				3		3		
PAL1.AW-->STL	3		2,10, 12,20			10	10		4,5		
PO1.PAL1-->TOK1	4			6					6		#
PO1.PAL1-->TOK2	5				7				7		#
PO1.TOK1*	6			8					8		
PO1.TOK2*	7				9				9		
PO1.TOK1-->PAL1	8			4,14					10		#
PO1.TOK2-->PAL1	9				5,15				10		#

Element		MAG	AW	TOK1	TOK2	STL	PAL 1	PAL 2	PO1	PO2	RO-BOT
Czynnosc	k / j	1	2	3	4	5	6	7	8	9	10
PAL1.STL-->AW	10		11			3.13	11		11		
PAL1.AW-->MAG	11	2.11. 12.21	2.10. 12.20				22		22		
PAL2.MAG-->AW	12	2.11. 12.21	13					13		13	
PAL2.AW-->STL	13		2.10. 12.20			20		20		14. 15	
PO2.PAL2-->TOK1	14			16						16	#
PO2.PAL2-->TOK2	15				17					17	#
PO2.TOK1*	16			18						18	
PO2.TOK2*	17				19					19	
PO2.TOK1-->PAL2	18			4.14				20			#
PO2.TOK2-->PAL2	19				5.15			20			#
PAL2.STL-->AW	20		21			3.13		21		21	
PAL2.AW-->MAG	21	2.11. 12.21	2.10. 12.20					22		22	
END	22						1	1	1	1	

OS(10.k) = {4.5.8.9.14.15.18.19}, k = 1.4.5.8.9.14.15.18.19

Przykładowo po zakończeniu czynności j = 3 (PAL1. AW-->STL) element k = 6 jest przygotowany do rozpoczęcia jednej z dwóch czynności : j = 4 (PO1. PAL1-->TOK1) lub j = 5 (PO1. PAL1-->TOK2). W tym przypadku Out(3.6) jest dwuelementowym zbiorem {4.5}.

Symulacja systemu opiera się na analizie systemu w kolejnych jego stanach. Liczba elementów k. dostępnych do rozpoczęcia zdarzenia j na etapie i jest określona macierzą $[N_{jk}(i)]$. Macierz stanu $[S_{jk}(i)]$ jest określona wzorem :

$$[S_{jk}(i)] = [N_{jk}(i)] - [U_{jk}] \qquad (1)$$

i podaje nadwyżkę elementów k dostępnych do rozpoczęcia czynności j na etapie i nad minimalną ich liczbą używaną w tej czynności. Macierz stanu $[S_{jk}(i)]$ oraz czas bieżący T(i) są wykorzystywane do wyznaczania początku lub końca czynności na etapie i. Stosownie do ostatniego zdarzenia oblicza się macierz stanu $[S_{jk}(i+1)]$ oraz czas T(i+1) dla następnego etapu (i+1).

Jeżeli na danym etapie i następuje rozpoczęcie czynności l . to wówczas liczba elementów k dostępnych do rozpoczęcia czynności j zmniejszy się wg wzoru :

$$S_{jk}(i+1) = S_{jk}(i) - U_{lk} + E_{jk}(i) \qquad (2)$$

Jeżeli na etapie i następuje zakończenie czynności l, to wówczas liczba elementów k dostępnych do rozpoczęcia czynności j zwiększy się wg wzoru :

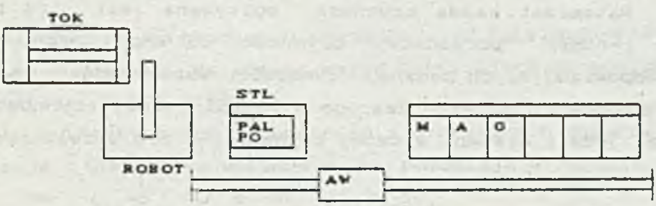
$$S_{jk}(i+1) = S_{jk} + U_{lk} * Out_{jk}(l) \qquad (3)$$

Macierze $[E_{jk}(l)]$ and $[Out_{jk}(l)]$ są wyznaczane algorytmicznie przez program obliczeniowy na podstawie zbiorów wyjść Out(j,k), zadeklarowanych przez użytkownika. $E_{jk}(l) = 1$ oznacza, że przy rozpoczęciu czynności l użyto elementu k ,który uprzednio mógł być użyty do rozpoczęcia czynności j. $E_{jk}(l) = 0$ oznacza, że przy rozpoczęciu czynności l nie użyto elementu k , który bierze udział w czynności j. Element macierzy $Out_{jk}(l) = 1$, jeżeli po zakończeniu czynności l element k jest przygotowany do rozpoczęcia czynności j, w przeciwnym przypadku $Out_{jk}(l) = 0$.

3. Algorytm przygotowania danych

Dla uniknięcia czasochłonnego rozpisywania identycznych elementów modelowanego systemu, powtarzających się sekwencji analogicznych czynności oraz odpowiadających im zbiorów wyjść, opracowano uproszczony algorytm przygotowania danych dla programu symulacyjnego.

Z modelowanego systemu (Rys.1) wyodrębnia się podsystem podstawowy i dla niego definiuje się zbiory wyjść. System ten tworzą pojedyncze elementy każdego rodzaju (Rys.2). W tablicy III przedstawiono zbiory wyjść dla podsystemu podstawowego.



Rys.2. Podsystem podstawowy Fig.2. Basic subsystem

Tablica III

Zbiory wyjść dla podsystemu podstawowego (Rys.2.)

Element		MAG	AW	TOK	STL	PAL	PO	ROBOT
Czynność	k / j	1	2	3	4	5	6	7
START	1	2	2	4	3	2	2	4,6
PAL,MAG-->AW	2	2,8	3			3	3	
PAL,AW-->STL	3		2,7		7	4	4	
PO,PAL-->TOK	4			5		6	5	4,6
PO,TOK*	5			6			6	
PO,TOK-->PAL	6			4		7	9	4,6
PAL,STL-->AW	7		8		3	8		
PAL,AW-->MAG	8	2,8	2,7			9		
END	9					1	1	

Zmniejszenie liczby elementów branych pod uwagę podczas tworzenia zbioru danych pozwoliło na znaczne ograniczenie liczby czynności elementarnych (w przypadku omawianego systemu ilość czynności elementarnych zmniejszyła się z 22 do 9, w przypadku modelowania większych systemów to uproszczenie jest jeszcze większe).

Następnie elementy podsystemu podstawowego zostają pogrupowane według podobieństwa zachowania się w systemie. Każdej grupie przypisany zostaje kolor. Obiektom, które w modelowanym systemie (Rys.1.) występują pojedynczo, przypisany zostaje kolor neutralny c_o. Tablica IV przedstawia podział na grupy (kolory) elementów podsystemu podstawowego.

Tablica IV

Podział na grupy (kolory)

kolor	c_o	c_1	c_2
obiekty	MAG, AW STL, ROBOT	TOK	PO PAL
liczba obiektów	1	2	2

Obecnie każdy element może zostać opisany za pomocą uporządkowanej pary liczb (k,c), gdzie k jest numerem porządkowym elementu, a "c" przypisanym mu kolorem. Natomiast każda czynność opisywana jest za pomocą pary $(j,\bar{c})$, gdzie j-numer porządkowy czynności ,a $\bar{c}(c_1,\ldots,c_m,\ldots,c_M)$-wektor kolorów odpowiadających zadanej czynności. Współrzędna c_m wektora $\bar{c}$ może być kolorem elementu uczestniczącego w rozpatrywanej czynności lub kolorem neutralnym, jeżeli element w danej czynności j nie uczestniczy.

Tablica

Wektory kolorów dla wybranych czynności modelowanego systemu

Wektor koloru	Przykładowe czynności
101	PAL1.AW-->STL
102	PAL2,AW-->STL
011	PO1.TOK1*
012	PO2.TOK1*
021	PO1.TOK2*
022	PO2.TOK2*

Wykorzystując tak zdefiniowane wektory kolorów dla systemu podstawowego generuje się wektory (Tablica V) kolorów i odpowiadające im nazwy czynności dla modelowanego systemu (jeżeli w czynności uczestniczy pojedynczy element, to jego numeracja porządkowa zostaje zaniechana).

Czynności, w których uczestniczą pojedyncze elementy lub elementy z indeksami "1", mają swoje bezpośrednie odpowiedniki w systemie podstawowym.

Ponieważ zwiększeniu uległa liczba rozpatrywanych obecnie elementów i czynności (odpowiednio z 7 do 10 i 9 do 22), konieczna jest modyfikacja [4] zadeklarowanych wcześniej zbiorów wyjść (zbiory te po modyfikacji zostaną przeniesione bezpośrednio do modelowanego systemu).

Renumeracja zbioru czynności elementarnych odbywa się wg zależności:

$$j^z \in \{1,\dots,J\} \cup \{j^d\} \qquad (4)$$

$$j^z \in \{1,\dots,j^d,\dots,J^z\}$$

dla czynności:

$$j < j^d \qquad j^z = j$$

$$j = j^d \qquad j^z = j^d$$

$$j > j^d \qquad j^z = j+1$$

gdzie:

j^d - nowo wprowadzona czynność,

j^z - czynność zmodyfikowana.

Analogiczne zmiany występują w zbiorze podstawowych elementów systemu $k \in \{1,\dots,K\}$.

Zmiany w strukturze symulowanego systemu pociągają za sobą zmiany w zbiorach $Out(j,k)$:

$Out(j^z,k)$ - po modyfikacji liczby czynności elementarnych.

$Out(j,k^z)$ - po modyfikacji liczby elementów systemu.

Przykładowo, po wprowadzeniu czynności $j=3$ PAL1.MAG-->AW (pobranie palety drugiej z magazynu na wózek) wyjść Out(1,8)=(2,8) zostanie zmodyfikowany do postaci: Out(1,9)=(2,8).

W celu automatycznego wygenerowania brakujących zbiorów wyjść definiuje się relacje $\bar{c}_l COR \bar{c}_j$. Dwa wektory $\bar{c}_l$ i $\bar{c}_j$ pozostają w relacji COR, jeżeli:

$$\forall m : c_{lm} = c_{jm} \cup c_{jm} = c_o \cup c_{lm} = c_o \qquad (5)$$

Brakujące wyjścia definiowane są wg zależności :

$$(l,\bar{c}) \in OS((j,\bar{c}),(k,c)) \text{ if } l \in OS(j,k) \text{ and } \bar{c}_l \, COR \, \bar{c}_j . \qquad (6)$$

Po dokonaniu tej operacji dla wszystkich elementów i czynności systemu podstawowego na podstawie danych tabeli III i IV automatycznie tworzone są dane zawarte w tabeli II. Jest to kompletny zbiór wyjść dla modelowanego systemu.

W opracowanym algorytmie przewidziano możliwość sumowania zbiorów wyjść dla czynności podobnych. Warunkiem koniecznym tego sumowania jest, by po zakończeniu zadanej czynności element przechodził w stan oczekiwania. Warunkiem wystarczającym, by element, dla którego zachodzi sumowanie, różnił się kolorem od pozostałych elementów uczestniczących w czynności.

4. Opis programu komputerowego

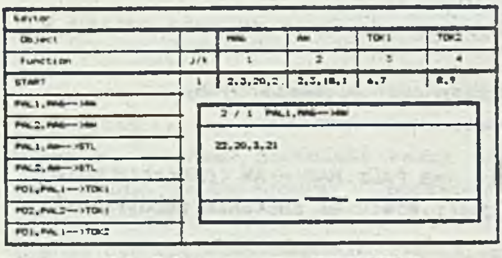
Na bazie przedstawionego algorytmu opracowano odpowiedni program komputerowy (program został zrealizowany w języku C).

Po zdefiniowaniu zbiorów wyjść dla elementów i czynności systemu podstawowego (Rys.3) określana jest liczebność poszczególnych elementów w modelowanym systemie (Rys.4.) oraz ich podział na grupy (kolory). Nazwy elementów o tym samym kolorze oddzielone są od siebie średnikami. W przypadku, gdy w danej grupie znajdują się tylko elementy identyczne (np. tokarki), przypisanie koloru odbywa sie automatycznie. Po wprowadzeniu powyższych danych program samoczynnie tworzy właściwe nazwy elementów i czynności oraz odpowiadające im zbiory wyjść (Rys.5).

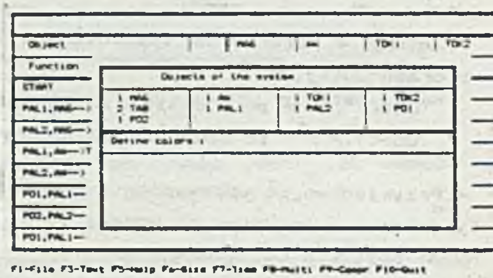
Dodatkowo w programie przewidziano możliwość modyfikacji już istniejących zbiorów danych. Modyfikacje te mogą polegać na:

- tworzeniu nowych zbiorów wyjść,
- modyfikacji zbiorów już istniejących,
- dopisywaniu do systemu nowych czynności i elementów bądź ich usuwaniu.

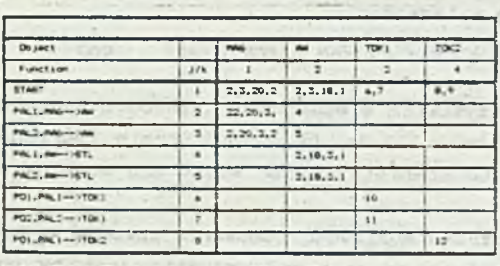
Zaletą programu jest przeprowadzanie podstawowej diagnostyki błędów dla tworzonych plików danych.



Rys.3. Zbiory wyjść dla systemu podstawowego (fragment)
Fig.3. Output sets for basic subsystem (fragment)



Rys.4. Definiowanie liczby elementów i wektorów kolorów
Fig.4. Defining number of objects and vectors of colors



Rys.5. Zbiory wyjść dla systemu modelowanego (fragment)
Fig.5. Output sets for modelled system (fragment)

5. Podsumowanie

Przedstawiony w referacie model macierzowy oraz algorytmiczny sposób przygotowania danych wykazują pełną analogię do kolorowych sieci Petriego. Ich zaletą jest jednak bardziej zwięzły i przystępny dla przeciętnego użytkownika zapis. Opracowany program komputerowy znacznie ułatwia i przyspiesza tworzenie i modyfikacje zbiorów danych dla programu symulacyjnego. Jest on szczególnie przydatny w przypadku przygotowania danych dla symulacji dużych i złożonych elastycznych systemów produkcyjnych.

LITERATURA

[1] Cyklis J.: Algorytm symulacji ESP. Zeszyty Naukowe Politechniki Śląskiej, s. Automatyka, z.96. Gliwice 1988.

[2] Cyklis J.: Pierzchała W.: Simulation and Control of FMS on its Operation Level. 6th Symposium on Information Control Problems in Manufacturing Technology. INCOM'89. Madrid. 1989.

[3] Cyklis J.: Symulacja elastycznych systemow produkcyjnych z wykorzystaniem macierzy stanu. Zeszyty Naukowe Pol.Slaskiej, s.Automatyka, z.85, Gliwice 1986.

[4] Czuła R.: Przygotowanie danych dla programu symulacyjnego ESP. Czasopismo Techniczne, Politechnika Krakowska (przyjęte do druku).

Recenzent: Prof.dr inż. Henryk Kowalowski

Wpłynęło do Redakcji do 30.04.1992 r.

Abstract:

If many objects of the same type are involved in the system much time is needed for putting the data into the simulation program. In order to solve that problem a specific model and the computer program has been worked out.

The so called objects (entities) colors are introduced $(c_1,\dots,c_m,\dots,c_M)$. As a matter of fact they are simply additional numbers of the objects having the same character but the name "color" shows that some analogy to the colored Petri nets [5] can be seen. The colors are declared if the number of objects of the same type is greater than one. The output set for single objects $OS(j,k)$ is declared. Now each entity of the system can be described by the pair (k,c), where c is the color of the object. If the object is only one $c=c_o$ (c_o - neutral color c)or the color can be simply omitted. The activity can be described by the pair $(j,\bar{c})$ where $\bar{c}(c_1,\dots,c_m,\dots,c_m)$ is the vector of colors. The coordinate c_m can be the color of object taking part in the activity j or neutral color c_o if the object does not take part in the activity j.

The relation $\bar{c}_l COR\ \bar{c}_j$ is defined. The two vectors $\bar{c}_l$ and $\bar{c}_j$ are in the relation COR if for each $m=1,\dots,M$: $c_{lm}=c_{jm} \cup c_{jm}=c_o \cup c_{lm}=c_o$.

The output set is derived in the following way:

$(1,\bar{c}) \in OS((j,\bar{c}),(k,c))$ if $1 \in OS(j,k)$ and $\bar{c}_l COR\ \bar{c}_j$.

The new numeration can be introduced $(k,c) \to k^n$ and $(j,\bar{c}) \to j^n$. After all these operations we received input data for the modelling system.