

## NEURONAL COMPUTER SYSTEM FOR FORECAST AMMONIA EMISSION AFTER APPLIED LIQUID MANURE

### Summary

The origin of undertaking this subject is related with the deep changes observed in manure management in Poland during last years. This is caused by intensification of animal production and changing the type of production from solid to liquid manure, which is unfavourable for agro-ecosystems. This situation is a base for creation of tool permitting the estimation of ammonia emission level during applied liquid manure. By identification of the main factors raising the emission level, this tool will be used also for reduction of  $NH_3$  losses.

## NEURONOWY SYSTEM KOMPUTEROWY PROGNOZUJĄCY POZIOM EMISJI AMONIAKU PO NAWOŻENIU GNOJOWICĄ

### Streszczenie

Genezą podjęcia tematu są głębokie zmiany zachodzących w gospodarce nawozami naturalnym w ostatnich latach w Polsce. Wiąże się to zwłaszcza z intensyfikacją produkcji i przestawianiem się wielu gospodarstw z hodowli obornikowej na gnojowicową, niekorzystną dla środowiska naturalnego. Sytuacja ta jest powodem opracowania narzędzia pozwalającego na oszacowanie poziomu emisji amoniaku do atmosfery podczas nawożenia gnojowicą. Dzięki identyfikacji głównych czynników zwiększających emisję, narzędzie to będzie przydatne również do redukcji jej wielkości.

### Wstęp

Celem niniejszej pracy było zaprojektowanie, wykonanie i przetestowanie, zgodnie ze standardami inżynierii oprogramowania, systemu informatycznego wykorzystującego technikę sztucznych sieci neuronowych, wspomagającego modelowanie poziomu strat azotu, spowodowanych emisją amoniaku w czasie nawożenia pól gnojowicą. Uzasadnieniem wyboru sztucznych sieci neuronowych do realizacji projektu są ich zdolności predykcyjne i aproksymacyjne oraz zdolność modeli neuronowych do generalizacji wiedzy nabytej podczas uczenia.

### Przygotowanie zbioru danych empirycznych i proces uczenia sieci neuronowej

Uzyskanie satysfakcjonujących parametrów jakościowych sieci neuronowej zdeterminowane jest posiadaniem reprezentatywnego zbioru danych empirycznych, manifestujących modelowane zjawisko [5]. Rys. 1 przedstawia fragment proponowanego zbioru uczącego. Cztery pierwsze kolumny zawierają dane wejściowe. Są to czynniki agrotechniczne mające znaczący wpływ na poziom emisji amoniaku z pól nawożonych gnojowicą. Wybrane one zostały w oparciu o doświadczenia przeprowadzone w Instytucie Inżynierii Rolniczej AR oraz przeprowadzone studia literaturowe.

- Początkowe stężenie azotu amonowego ( $N-NH_4$ ) w nawozie. Podczas doświadczeń użyta została gnojowica o stężeniu od 1.5 do 6.5 [g/kg].
- Dawka gnojowicy zastosowana na polu w trakcie doświadczenia. W przypadku doświadczeń użytych do kompletowania zbioru uczącego wahała się przedziale od 6 do 43  $m^3/ha$ .

- Okres agrotechniczny, w którym odbywało się doświadczenie, określony został jako kolejny tydzień roku. Nawożenie gnojowicą dozwolone jest tylko pomiędzy 1 marca a 31 listopada i na termin ten przypadają tygodnie od 9 do 43.
- Technika, jaką gnojowica rozlewana była po polu w trakcie danego doświadczenia. W wykorzystanych doświadczeniach gnojowica aplikowana była trzema najpopularniejszymi metodami: rozlew tradycyjny poprzez płytki rozbryzgowe, rozlew powierzchniowy poprzez przystawkę z węzami wleczonymi i rozlew doglebowy za pomocą aplikatorów doglebowych.

Piąta kolumna stanowi predykcyjna zmienna wyjściowa. Prognozowaną wielkością są straty azotu spowodowane emisją amoniaku do atmosfery. Przedstawiona jest ona w postaci procentowej utraty początkowej zawartości azotu amonowego ( $N-NH_4$ ) w nawozie. Zebranie reprezentatywnego zbioru uczącego polegało na analizie wyszukanych sprawozdań z badań i artykułów naukowych w internetowych bibliotekach: Elsevier i ASABE Technical Library, a następnie pozyskaniu właściwych danych. Tą drogą zgromadzono zbiór uczący zawierający 154 przypadki trenin-gowe.

Do wytworzenia topologii sztucznej sieci neuronowej zastosowany został komercyjny pakiet *Statistica Neural Network 4.0*. stanowiący zaawansowany symulator jednokierunkowych modeli neuronowych. Daje on użytkownikowi m.in. możliwość prognozowania ciągłej zmiennej liczbowej z określonego zakresu przy wykorzystaniu takich typów sieci jak perceptron wielowarstwowy *MLP*, sieci o radialnych funkcjach bazowych *RBF*, sieci regresyjne czy sieci liniowe [6].

Po odpowiednim przygotowaniu zbioru uczącego przeprowadzony został szereg testów, właściwych dla wszyst-

kich dostępnych typów sieci neuronowych. Najkorzystniejsze wartości parametrów mówiących o jakości sieci używały sieci *MLP (Multi Layer Perceptron)*. Posiadały one najlepszy współczynnik korelacji wartości błędów dla zbiorów walidacyjnego i testowego, co wskazuje na zdolność wygenerowanych sieci do generalizowania nabytej wiedzy [8]. Ostatecznie wybrana została trójwarstwowa sieć typu *MLP* z 15-oma neuronami w pierwszej warstwie ukrytej i 5-oma w drugiej.

Wytworzoną sieć wyeksportowano do programu *Statistica 6.0*, który posiada narzędzie *Generator kodu*, umożliwiające pozyskanie wygenerowanej wcześniej sieci neuronowej w postaci kodu języka C. Informacje niezbędne do prawidłowego działania sieci takie jak wartości wagowe i progowe zapisane są statycznie w postaci tablic liczb rzeczywistych o podwójnej dokładności [7].

## Implementacja kodu aplikacji *ProgAzot*

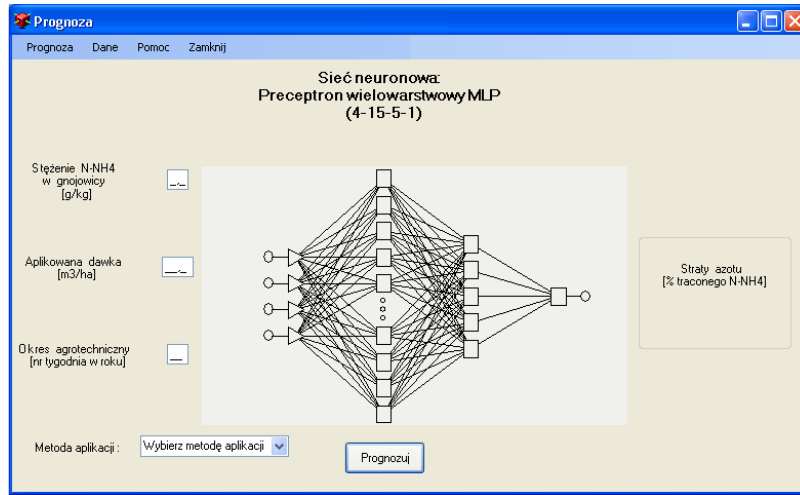
System prognozujący straty azotu po nawożeniu gnojowicą *ProgAzot* (Rys. 2) wytworzono wykorzystując środowisko programistyczne *Visual Studio .NET*. Jego istotną zaletą z punktu widzenia programisty, jest oddzielenie prowadzanego kodu od interfejsu użytkownika na poziomie tworzenia aplikacji. Daje to pełniejszą kontrolę nad wytwarzanym oprogramowaniem oraz ułatwia wprowadzenie ewentualnych zmian [4]. Kod sieci wygenerowany przez program *Statistica 6.0* musiał ulec pewnym modyfikacjom, wynikającym z ewolucji rodziny języków C, ponieważ aplikacja pisana była w języku *Visual C#* [3]. Po zaprojektowaniu systemu, wygenerowaniu kodu sieci i wyborze środowiska programistycznego kolejną fazą było wytworzenie poszczególnych modułów aplikacji.

	1	2	3	4	5
	Zawartość N-NH4 [g/kg]	Aplikowana dawka [m <sup>3</sup> /ha]	Okres agrotechniczny [-]	Metoda aplikacji	Emisja amoniaku [% TAN]
103	2.5	16.4	36	rozlewanie	67.8
104	2.1	17.3	11	rozlewanie	86.2
105	2.2	17.6	12	rozlewanie	84.8
106	1.8	19.1	16	doglebowo	5.2
107	1.8	17.9	16	doglebowo	2.8
108	1.8	19.2	16	doglebowo	3.8
109	1.8	18.7	16	rozlewanie	57.2
110	2.6	13.5	17	przystawka	30.1
111	2.6	14	17	przystawka	11.9

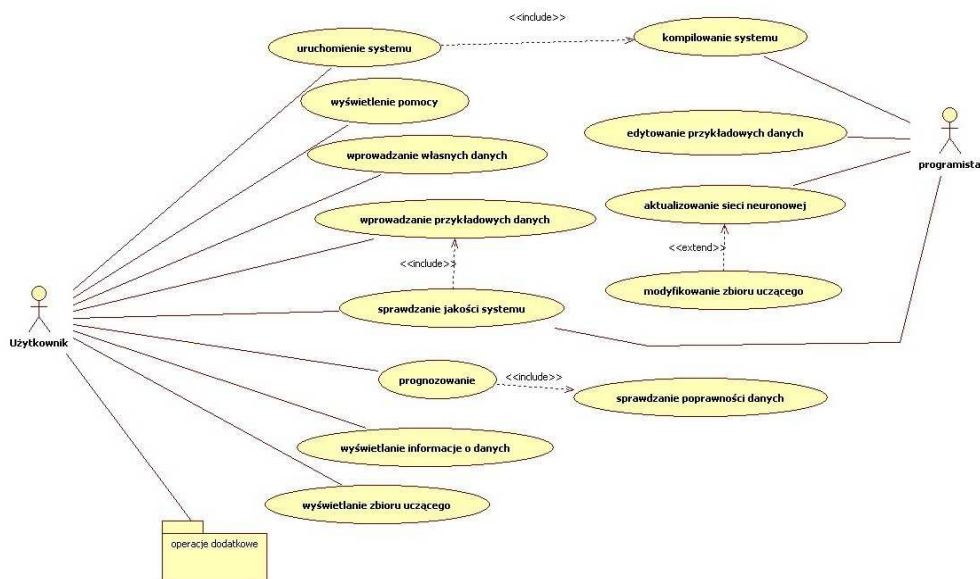
Rys. 1. Fragment zbioru danych do uczenia sieci neuronowej  
Fig. 1. Fragment of data for neural network learning



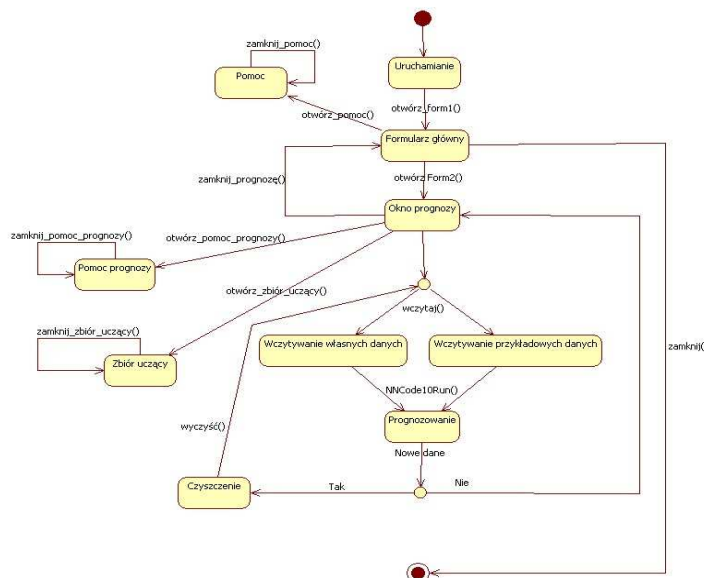
Rys. 2. Główne okno aplikacji *ProgAzot*  
Fig. 2. *ProgAzot* application main window



Rys. 3. Okno prognozy  
Fig. 3. Prognosis window



Rys. 4. Diagram przypadków użycia  
Fig. 4. Use case diagram



Rys. 5. Diagram maszyny stanów  
Fig. 5. State diagram

Podstawowe możliwości systemu *ProgAzot* to:

- prognozowanie strat azotu z pól nawożonych gnojowicą,
- testowanie jakości prognozy na bazie danych doświadczalnych,
- dostęp do pełnego zbioru danych w postaci arkusza *MS Excel*, który posłużył do wytrenowania sieci neuronowej, w celu ewentualnego wykorzystania go w symulatorze sztucznych sieci neuronowych,
- wyświetlenie pomocy dotyczącej wprowadzania danych do prognozy,
- pomoc programu.

Najważniejszy moduł reprezentuje okno prognozy (Rys. 3). Funkcja *NNCodeRun* zawiera kod *C#* sieci neuronowej. Realizuje ona prognozowanie na podstawie wartości czynników agrotechnicznych podanych na wejścia sieci neuronowej, w oparciu o przechowywane w tablicach wartości progowe i wagowe poszczególnych neuronów. Wytworzona została również osobna funkcja konwertująca wartości przesyłane pomiędzy funkcją *NNCodeRun* a interfejsem użytkownika.

Jednym z modułów aplikacji jest system pomocy wprowadzający użytkownika w podstawowe zagadnienia dotyczące sztucznych sieci neuronowych oraz w problematykę strat azotu i emisji amoniaku z pól nawożonych gnojowicą.

### Projektowanie systemu informatycznego

Efektywność pracy nad wytwarzaniem oprogramowania zdeterminowana jest właściwą organizacją wszystkich jej etapów. Kolejne jej fazy, począwszy od podjęcia decyzji o sposobie realizacji projektu, poprzez implementację aż do użytkowania, powinny być wykonywane według ustalonej systematyki [2]. Nadzór nad niniejszą aplikacją oparto o kaskadowy model życia oprogramowania, który jest klasycznym i najczęściej stosowanym modelem.

Fazie określania wymagań potencjalnych użytkowników aplikacji towarzyszył proces konstruowania jej architektury. Dokumentacja implementacji systemu zapisana została z zastosowaniem notacji graficznej UML [1]. Odzwierciedleniem wymagań funkcjonalnych jest diagram przypadków użycia (Rys 4), natomiast opis stanów i zdarzeń powodujących przejścia pomiędzy stanami przedstawia diagram maszyny stanów (Rys 5).

Rdzeniem tworzonego systemu informatycznego jest wytrenowana sztuczna sieć neuronowa, dlatego też od uzyskania jak najlepszych współczynników mówiących o jej jakości, uzależniona była walidacja całego systemu.

### Testowanie wytworzonej aplikacji *ProgAzot*

Testowanie programu przeprowadzane było podczas implementacji programu a także po wytworzeniu wszystkich modułów systemu. Celem testowania była detekcja, a następnie korekta ewentualnych błędów w kodzie aplikacji, ocena jej niezawodności oraz sprawdzenie czy spełnia zadania postawione przed nią w fazie określania wymagań klienta. Odbywało się ono na dwa sposoby: statyczny, sprowadzający się do analizy kolejnych instrukcji kodu i dynamiczny, polegający na porównaniu rezultatów otrzymanych po wykonaniu kolejnych fragmentów aplikacji z

wynikami oczekiwanymi. Ważnym i wysoce zaawansowanym narzędziem, umożliwiającym wykrycie nawet najbardziej ukrytych błędów w aplikacji jest, zawarty w środowisku *MS Visual Studio 2005, debugger*. Przeprowadza on obserwację przebiegu poszczególnych funkcji aplikacji i wyszukuje błędne fragmenty kodu oraz punkty wstrzymania [4].

### Wnioski

1. Wytworzony system komputerowy *ProgAzot* pozwala prognozować poziom wielkości strat azotu, spowodowanych emisją amoniaku w czasie nawożenia pól gnojowicą. Złożoność modelowanego zjawiska pozwala przyjąć, że średni błąd predykcji popełniany przez aplikację jest na akceptowalnym poziomie.
2. Na podstawie przeprowadzonych testów stwierdzono, iż techniki neuronowe są właściwym narzędziem generalizacyjnym, umożliwiającym wytworzenie modelu do prognozy strat azotu. Najkorzystniejsze wartości parametrów, mówiących o jakości wytrenowanej sieci, uzyskano dla sztucznej sieci neuronowej typu perceptron o topologii trójwarstwowej (4-15-5-1)
3. Opracowany system informatyczny spełnia założenia funkcjonalne postawione w fazie określania wymagań oraz wymogi inżynierii oprogramowania pod względem jakości i funkcjonalności.
4. Proponowany system informatyczny jest narzędziem umożliwiającym oszacowanie skali zagrożenia wzrastającą emisją amoniaku oraz jej symulację przy danych parametrach agrotechnicznych podczas nawożenia gnojowicą. Może być również z powodzeniem wykorzystana przez rolników, w celu doboru odpowiedniego terminu, dawki i metody rozlewu nawozu.

### Literatura

- [1] Booch G., Rumbaugh J., Jacobson I., 2000. Inżynieria oprogramowania UML – przewodnik użytkownika. Warszawa.
- [2] Jaszkievicz A., 1997. Inżynieria oprogramowania. Helion, Gliwice.
- [3] Liberty J., 2006. Visual C# 2005: Zapiski programisty. Helion, Gliwice.
- [4] Stefańczyk A., 2005. Sekrety języka C#. Internetowe Wydawnictwo Złote Myśli, Gliwice.
- [5] Tadeusiewicz R., 2000. Biocybernetyka i inżynieria biomedyczna. Tom 6: Sieci neuronowe. Exit, Warszawa.
- [6] Tadeusiewicz R., Lula P., 2001. Statistica Neural Networks PL: wprowadzenie do sieci neuronowych. StatSoft Polska, Kraków.
- [7] Wójtowicz P., 2001. Polska wersja Statistica Neural Networks. StatSoft Polska, Kraków.
- [8] Żurada J., Barski M., Jędruch W., 1996. Sztuczne sieci neuronowe. Wydawnictwo Naukowe PWN, Warszawa.
- [9] Boniecki P., Weres J., 2003. Wykorzystanie technik neuronowych do predykcji wielkości zbiorów wybranych płodów rolnych. Journal of Research and Applications in Agricultural Engineering, Poznań, vol. 48(4), str. 56-59