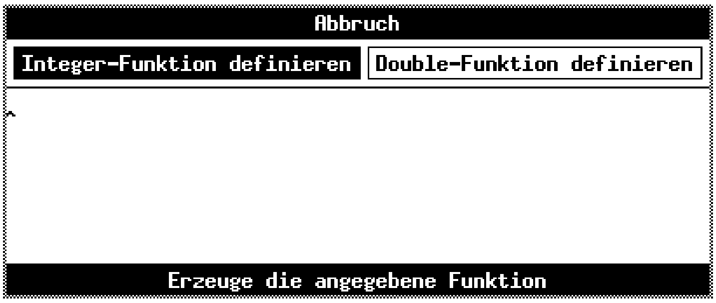

Abbildung 18.8: Das Popup–Fenster des Menüunterpunktes *Funktionen definieren*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.4 dargestellt ist, erscheint wieder.
- Die beiden Buttons Integer–Funktion definieren und Double–Funktion definieren dienen als Umschalter (Toggles) zwischen den beiden möglichen Funktionstypen. Wenn der Toggle Integer–Funktion definieren aktiv ist (wie in Bild 18.8), dann kann der Benutzer eine Integerfunktionsvariable erzeugen. Ist dagegen der Umschalter Double–Funktion definieren aktiv, so wird eine Gleitkommafunktionsvariable generiert. Die Umschaltung zwischen den beiden Alternativen wird folgendermaßen durchgeführt:
  - Den Mauszeiger auf den inaktiven Button setzen.
  - Durch Drücken der linken Maustaste wird der gewünschte Funktionstyp gesetzt.
- Mit dem Button Erzeuge die angegebene Funktion wird der darüberliegende Text als Funktion des vorliegenden Typs erzeugt:
  - Der eingegebene Funktionstext muß sich an folgende Syntax halten:

function_text	::=	**name**“*(*“ **var_list** “*)*“ “=“ **funct_expression.**
name	::=	**„Bezeichner, der noch nicht definiert ist“.**
var_list	::=	**var_name** { “,“ **var_name** }.
var_name	::=	**„Bezeichner (Funktionsvariable)“.**
funct_expression	::=	**term** { (“+“ | “–“) **term** }.
term	::=	**factor** { (“*“ | “/“) **factor** }.
factor	::=	**power** { „ˆ “ **power** }.
power	::=	[“–“] **scalar.**
scalar	::=	“*(*“ **funct_expression** “*)*“ | **function.**
function	::=	“*sin(*“ **funct_expression** “*)*“ | “*cos(*“ **funct_expression** “*)*“ | “*tan(*“ **funct_expression** “*)*“ | “*asin(*“ **funct_expression** “*)*“ | “*acos(*“ **funct_expression** “*)*“ | “*atan(*“ **funct_expression** “*)*“ |

		“sinh(“ **funct_expression** *“)“* |
		“cosh(“ **funct_expression** *“)“* |
		“tanh(“ **funct_expression** *“)“* |
		“sqrt(“ **funct_expression** *“)“* |
		“ln(“ **funct_expression** *“)“* |
		“log(“ **funct_expression** *“)“* |
		var_or_const |
		own_function |
		var_name.
var_or_const	::=	**„Eine Konstante oder Variable“**
own_function	::=	**„Eine vom Benutzer definierte Funktion (vom Typ int oder double).“**

* **name** ist der noch nicht definierte Bezeichner, mit dem die Funktion im System gespeichert wird.
* **var_list** ist die durch Kommata getrennte Liste der Funktionsargumente. Diese Namen sind nur lokal bei der Definition bekannt. D.h. es können beliebige — auch vordefinierte — Bezeichner verwendet werden; die ursprüngliche Bedeutung wird kurzfristig verdeckt!
* Für **own_function** gilt die gleiche Syntax wie bei einer Funktionsauswertung; es dürfen aber zusätzlich noch alle in der Definition vorkommenden Funktionsargumente verwendet werden!
* Bei der Definition können die beiden Datentypen *int* und *double* beliebig miteinander vermischt werden.
* Indem z.B. Operationen auf Konstanten zusammengefaßt werden, versucht ProSiNN **funct_expression** zu vereinfachen. Dabei werden aber Teilausdrücke nicht permutiert, so daß eine lokale und keine globale Vereinfachung stattfindet.
* Bei der Erzeugung einer Integerfunktion wird von allen vorkommenden Konstanten vom Typ *double* der Nachkommaanteil abgeschnitten. Analog gilt bei einer Funktionsauswertung innerhalb der Definition, daß alle Teilfunktionen vom Typ *double* und deren Werte intern auf den nächsten kleineren Integer abgebildet werden (bei negativen Zahlen werden sie auf den nächsten größeren Integer transformiert).

– Paßt der eingegebene Text nicht mehr in das in Abbildung 18.8 dargestellte Texteingabefenster, so erzeugt ProSiNN ein vertikales und/oder ein horizontales Scrollfeld. Mit ihnen kann der Benutzer jeden Teil des Funktionstextes erreichen.

Entdeckt das System einen Fehler im Eingabetext, so erscheint der invertierte Fehlerausgabebutton (mit der Fehlermeldung), der über dem Ein/Ausgabefenster plaziert wird (siehe Unterabschnitt 3.4.2). Nachdem der Anwender den Button mit einer Maustaste angeklickt hat, ist die Fehlerstelle im Eingabefenster der Funktion invertiert dargestellt.

Wurde die Funktionsvariable korrekt erzeugt, so erscheint wieder die Auswahlbox, die in Abbildung 18.4 dargestellt ist.

18.3.5 Der Menüunterpunkt „Benutzerkommandos definieren“

Mit dem Button Benutzerkommandos definieren kann der Benutzer ein Makro erzeugen. Nach dessen Auswahl erscheint das Popup–Fenster, das in Bild 18.9 dargestellt ist:

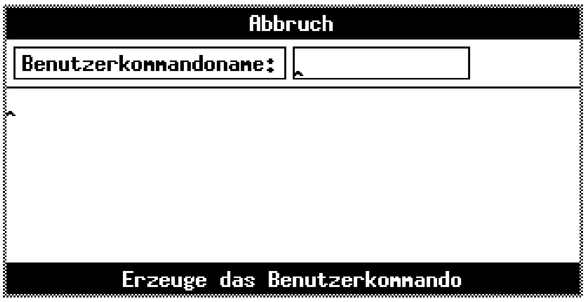

Abbildung 18.9: Das Popup–Fenster des Menüunterpunktes *Benutzerkommandos definieren*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.4 dargestellt ist, erscheint wieder.
- Der Bezeichner, den das zu erzeugende Makro erhalten soll, wird wie folgt eingegeben:
  - Den Mauszeiger auf das Eingabefeld für den Benutzerkommandonamen setzen.
  - Mit der Tastatur einen gültigen (siehe Abschnitt 4.1) noch nicht definierten Bezeichner eingeben.
- Mit dem Button Erzeuge das Benutzerkommando wird der darüberliegende Text als Makro erzeugt:
  - Der eingegebene Kommandotext muß sich an folgende Syntax halten:

    macro_text ::= command *“;“* { **command** *“;“* }.
    command ::= „Ein gültiger ProSiNN–Befehl“.

    * **command** bezeichnet ein beliebiges von ProSiNN unterstütztes Kommando. Auch der Name eines anderen bereits generierten Makros ist zulässig.
    * Die Korrektheit der Anweisungen wird erst beim ersten Aufruf des Makros überprüft!
    * Alle im Anweisungsblock neu erzeugten Variablen sind auch nach Ausführung des Makros noch definiert. Es können also keine lokalen Variablen generiert werden.
  - Paßt der eingegebene Text nicht mehr in das in Bild 18.9 dargestellte Eingabefenster, so erzeugt ProSiNN ein horizontales und/oder ein vertikales Scrollfeld. Mit ihnen kann jeder Teil des Makrotextes erreicht werden.

Wurde ein Makro erzeugt, so erscheint wieder die Auswahlbox, die in Abbildung 18.4 dargestellt ist.

18.3.6 Der Menüunterpunkt „Netzvariablen definieren“

Mit dem Button Netzvariablen definieren kann der Anwender Variablen vom Typ *layer*, *unit* oder *weight* erzeugen. Dieser Button ist nur dann in der Abbildung 18.4 vorhanden, wenn ein Neuronales Netz im System vorliegt. Nach dessen Auswahl erscheint z.B. das Popup–Fenster, das in Bild 18.10 dargestellt ist:

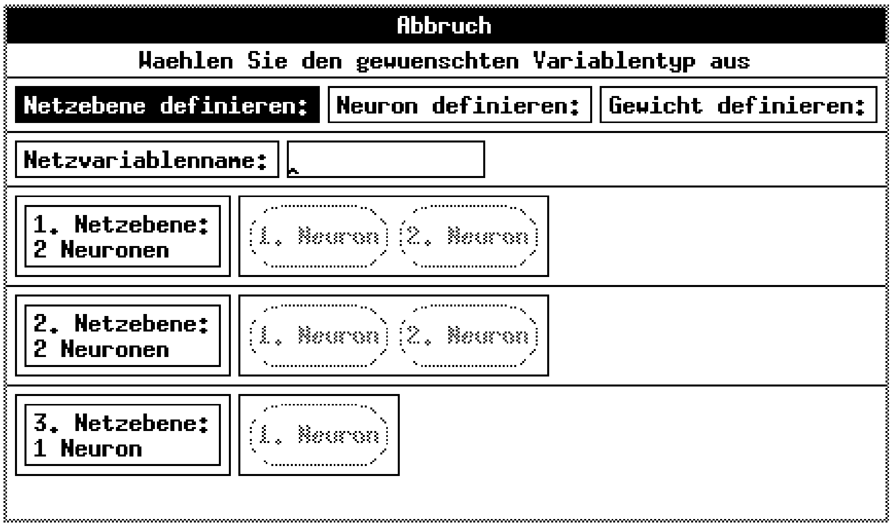

Abbildung 18.10: Das Popup–Fenster des Menüunterpunktes *Netzvariablen definieren*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.4 dargestellt ist, erscheint wieder.
- Die Buttons Netzebene definieren, Neuron definieren und Gewicht definieren dienen als Umschalter (Toggles) zwischen den möglichen Variablentypen. Wenn der Toggle Netzebene definieren aktiv ist (wie in Bild 18.10), dann kann der Benutzer eine Variable vom Typ *layer* erzeugen. Ist dagegen Neuron definieren aktiviert, so wird eine Neuronenvariable generiert. Wenn der Umschalter Gewicht definieren aktiv ist, so kann eine Variable vom Typ *weight* erzeugt werden. Die Umschaltung zwischen den Alternativen wird folgendermaßen durchgeführt:
  - Den Mauszeiger auf den gewünschten inaktiven Button setzen.
  - Durch Drücken der linken Maustaste wird der Variablentyp gesetzt.
- Ist das Netz zu groß für das in Abbildung 18.10 dargestellte Fenster, so erzeugt ProSiNN ein horizontales und ein vertikales Scrollfeld. Mit ihnen kann der Anwender jede Strukturkomponente erreichen.
- Der Bezeichner, den die zu erzeugende Variable erhalten soll, wird wie folgt eingegeben:
  - Den Mauszeiger auf das Eingabefeld für den Netzvariablennamen setzen.

  - Mit der Tastatur einen gültigen (siehe Abschnitt 4.1) noch nicht definierten Bezeichner eingeben.

- Eine Variable wird generiert, wenn sich der Benutzer in der dargestellten Netzstruktur je nach Toggle an eine der nachfolgenden Alternativen hält:
  - Soll eine Netzebenenvariable definiert werden, so muß der Anwender mit der linken Maustaste die gewünschte Schicht anwählen.
  - Möchte man eine Neuronenvariable definieren, dann muß mit der linken Maustaste die gewünschte aktive Zelle ausgewählt werden.
  - Soll eine Gewichtsvariable definiert werden, so muß man zuerst mit der linken Maustaste das gewünschte aktive Startneuron setzen. Anschließend sind alle Zellen inaktiv, die nicht in der nachfolgenden Ebene liegen. Nun muß mit der Maus das Zielneuron angewählt werden.

  Wurde die Variable korrekt erzeugt, so wird der Variablennamen gelöscht und der Anwender kann weitere Netzvariablen generieren.

  Konnte die Variable nicht erzeugt werden, dann wird eine Fehlermeldung ausgegeben und der Benutzer kann seine Eingaben korrigieren.

18.4 Der Menüpunkt „Variablen bearbeiten“

Will der Anwender einzelne Variablen (oder Kommandomakros) freigeben oder deren Werte verändern, dann muß er den Menüpunkt *Variablen bearbeiten* anwählen. Danach erscheint z.B. das Popup–Fenster, das in Bild 18.11 dargestellt ist[1].

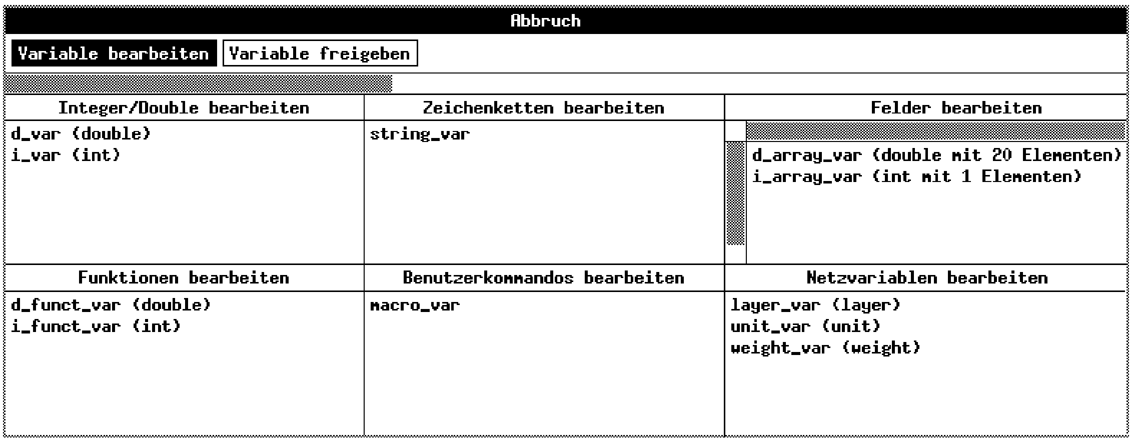

Abbildung 18.11: Die Auswahlbox des Menüpunktes *Variablen bearbeiten*

Bem.:

- Mit dem Button Abbruch kann der Menüpunkt beendet werden.

[1] Bei dieser Abbildung handelt es sich um eine Bildmontage. Normalerweise stellt das erzeugte Fenster nur einen Teil des hier sichtbaren Bereichs dar

- Die beiden Buttons [Variable bearbeiten] und [Variable freigeben] dienen als Umschalter (Toggles) zwischen den möglichen Bearbeitungsmodi. Wenn der Toggle [Variable bearbeiten] aktiv ist (wie in Abbildung 18.11), dann kann der Benutzer eine beliebige in den sechs Teilfenstern angegebene Variable betrachten oder ändern. Ist dagegen [Variable freigeben] aktiv, so wird eine Variable aus dem System entfernt, wenn sie mit dem Mauszeiger angeklickt wird. Die Umschaltung zwischen den beiden Alternativen wird folgendermaßen durchgeführt:
  - Den Mauszeiger auf den inaktiven Button setzen.
  - Durch Drücken der linken Maustaste wird der gewünschte Bearbeitungsmodus gesetzt.
- ProSiNN stellt in der in Bild 18.11 gezeigten Auswahlbox sechs Teilfenster mit den verschiedenen Variablentypen zur Verfügung:
  - In jedem Teilfenster werden die Variablen alphabetisch angeordnet.
  - Hinter jedem Variablennamen steht — wenn notwendig — in Klammern der Typ der Variablen und bei einem Feld zusätzlich noch die Anzahl der Elemente.
  - Passen nicht alle Variablen eines Typs in das dazugehörige Teilfenster, so erzeugt ProSiNN horizontale und vertikale Scrollfelder. Mit ihnen kann der Anwender jede Variable erreichen.
- Zum Freigeben einer Variable muß der Benutzer wie folgt vorgehen:
  - Den Mauszeiger auf den gewünschten Variablennamen setzen.
  - Durch Drücken der linken Maustaste wird die angegebene Variable aus der vorliegenden Liste und dem System entfernt.
- Will man den Inhalt einer Variablen betrachten oder ändern, so muß folgendermaßen vorgegangen werden:
  - Den Mauszeiger auf den gewünschten Variablennamen setzen.
  - Durch Drücken der linken Maustaste wird für die angegebene Variable ein weiteres Popup–Fenster erzeugt, in der die Variable betrachtet oder deren Wert gändert werden kann. Die Beschreibung dieser Fenster wird in den folgenden Unterabschnitten vorgenommen.

18.4.1 „Integer/Doubles bearbeiten“

Wenn der Benutzer mit dem Mauszeiger eine Variable vom Typ *int* oder *double* zum Bearbeiten ausgewählt hat, dann erscheint z.B. das Popup–Fenster, das in Abbildung 18.12 dargestellt ist (hier wurde die Variable `d_var`, die in Bild 18.11 abgebildet ist, ausgewählt).

Bem.:

- Durch Anklicken des Buttons [Abbruch] wird das Fenster beendet und die Auswahlbox, die in Bild 18.11 dargestellt ist, erscheint wieder.

Abbildung 18.12: Das Popup–Fenster zum *Integer/Doubles bearbeiten*

- Will der Anwender den Wert der dargestellten Variablen ändern, so muß er folgendermaßen vorgehen:
  - Den Mauszeiger auf das Eingabefeld für den Wert setzen.
  - Den dort angegebenen Wert auf die gewünschte Zahl korrigieren, die mit dem vorliegenden Variablentyp konform sein muß.
- Nachdem der Benutzer die Modifikation vorgenommen hat, kann der Wert durch Anklicken des Buttons Gewuenschten Wert setzen an die Variable zugewiesen werden.

  Konnte die Zuweisung durchgeführt werden, dann bleibt der eingegebene Wert unverändert und der Anwender kann das Fenster mit dem Button Abbruch beenden.

  Ist die eingegebene Zahl nicht korrekt, so wird der letzte gültige Wert der Variablen wieder im Eingabefenster dargestellt.

18.4.2 „Zeichenketten bearbeiten“

Wenn der Benutzer mit dem Mauszeiger eine Variable vom Typ *string* zum Bearbeiten ausgewählt hat, dann erscheint z.B. das Popup–Fenster, das in Abbildung 18.13 dargestellt ist (hier wurde die Variable `string_var`, die in Bild 18.11 abgebildet ist, ausgewählt):

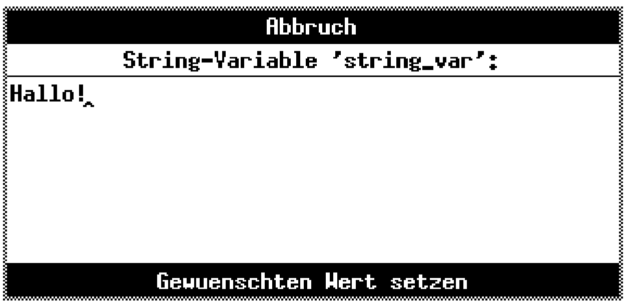

Abbildung 18.13: Das Popup–Fenster zum *Zeichenketten bearbeiten*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.11 dargestellt ist, erscheint wieder.

- Will der Anwender den Wert der dargestellten Variablen ändern, so muß er folgendermaßen vorgehen:
  - Den Mauszeiger auf das Eingabefeld für den Text setzen.
  - Jetzt kann die vorliegende Zeichenkette beliebig modifiziert werden (auch das Zeilenendezeichen und der leere String sind erlaubt).
  - Paßt der Text nicht mehr in das in Abbildung 18.13 dargestellte Eingabefenster, so erzeugt ProSiNN ein horizontales und/oder ein vertikales Scrollfeld. Mit ihnen kann jeder Teil der Zeichenkette erreicht werden.
- Nachdem der Benutzer die Modifikation vorgenommen hat, kann der Wert durch Anklicken des Buttons Gewuenschten Wert setzen an die Variable zugewiesen werden.

  Anschließend wird das Fenster mit Hilfe des Abbruch–Buttons wieder verlassen.

18.4.3 „Felder bearbeiten“

Wenn der Anwender mit dem Mauszeiger eine Feldvariable vom Typ *int* oder *double* zum Bearbeiten ausgewählt hat, dann erscheint in der Regel das Popup–Fenster, das in Bild 18.14 abgebildet ist (hier wurde die Feldvariable `d_array_var` mit 20 Elementen, die in Abbildung 18.11 dargestellt ist, ausgewählt):

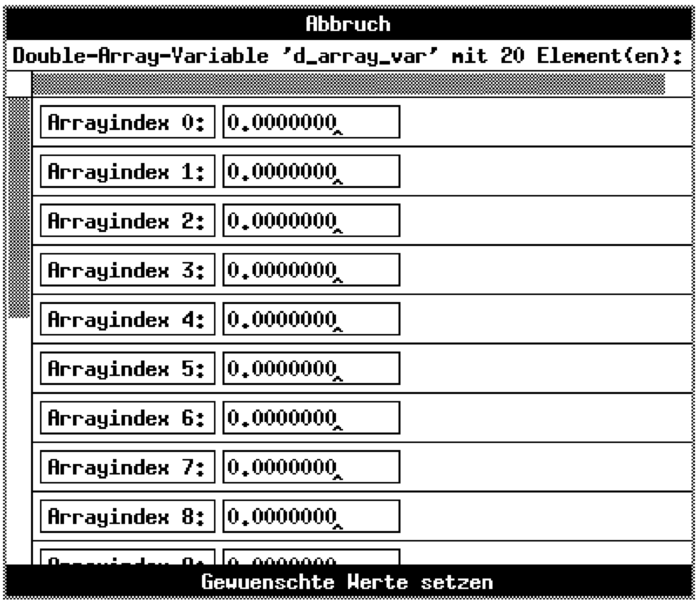

Abbildung 18.14: Das Popup–Fenster zum *Felder bearbeiten*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.11 dargestellt ist, erscheint wieder.

- Wenn die dargestellte Feldvariable maximal zehn Elemente enthält, werden keine Scrollfelder generiert. Hat sie dagegen mehr als zehn und weniger als 1001 Elemente, dann erzeugt ProSiNN Scrollfelder zum Erreichen aller Strukturkomponenten. Sollte die Variable mehr als 1000 Elemente enthalten, so wird ein spezielles Popup–Fenster generiert (siehe unten).

- Will der Benutzer einen internen Wert der dargestellten Feldvariablen ändern, so muß er folgendermaßen vorgehen:

  - Den Mauszeiger auf das Eingabefeld des zu ändernden Elementes setzen.
  - Den dort angegebenen Wert auf die gewünschte Zahl korrigieren, die mit dem vorliegenden Variablentyp konform sein muß.

- Nachdem der Anwender die Modifikationen vorgenommen hat, können die Werte durch Anklicken des Buttons Gewuenschte Werte setzen an die Variable zugewiesen werden.

  Konnten die Zuweisungen durchgeführt werden, dann bleiben die eingegebenen Werte unverändert und der Benutzer kann das Fenster mit dem Button Abbruch beenden.

  Ist eine eingegebene Zahl nicht korrekt, so wird der letzte gültige Wert des Elementes dargestellt. Diese Zahl kann nun erneut bearbeitet werden.

- Wenn der Anwender eine Feldvariable ausgewählt hat, die mehr als 1000 Elemente beinhaltet, dann wird z.B. das in Bild 18.13 dargestellte Popup–Fenster generiert:

Abbildung 18.15: Das Popup–Fenster für große zu bearbeitende Felder

  - Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.11 dargestellt ist, erscheint wieder.
  - Will der Benutzer einen internen Wert der dargestellten Feldvariablen ändern, so muß er folgendermaßen vorgehen:
    - Den Mauszeiger auf das Eingabefeld für den Arrayindex setzen.
    - Den dort angegebenen Wert auf den gewünschten Index korrigieren.
    - Durch Anklicken des Buttons Setze Arrayindex wird der Wert des entsprechenden Feldelementes angezeigt.
    - Um nun den Wert des angezeigten Elementes zu modifizieren, muß wie folgt vorgegangen werden:
      - Den Mauszeiger auf das Eingabefeld für den Wert setzen.
      - Den dort angegebenen Wert auf die gewünschte Zahl korrigieren, die mit dem vorliegenden Variablentyp konform sein muß.

∗ Nachdem der Anwender die Änderung vorgenommen hat, kann der Wert durch Anklicken des Buttons Gewuenschten Wert setzen an die Variable zugewiesen werden.
Konnte die Zuweisung durchgeführt werden, dann bleibt der eingegebene Wert unverändert und der Benutzer kann das Fenster mit dem Button Abbruch beenden oder ein anderes Feldelement bearbeiten.
Ist die eingegebene Zahl nicht korrekt, so wird der letzte gültige Wert dieses Elementes wieder im Eingabefeld dargestellt.

18.4.4 „Funktionen bearbeiten“

Wenn der Anwender mit dem Mauszeiger eine Funktionsvariable vom Typ *int* oder *double* zum Bearbeiten ausgewählt hat, dann erscheint das Popup–Fenster, das in Bild 18.16 abgebildet ist (hier wurde die Funktionsvariable `d_funct_var`$(x) = \sin(x)$, die in Abbildung 18.11 dargestellt ist, ausgewählt):

Abbildung 18.16: Die Auswahlbox zum *Funktionen bearbeiten*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.11 dargestellt ist, erscheint wieder.
- Durch Auswahl des Buttons Funktionswerte berechnen wird ein weiteres Popup–Fenster erzeugt, das z.B. die Gestalt von Bild 18.17 aufweist:

Abbildung 18.17: Das Popup–Fenster zum *Funktionswerte berechnen*

  – Mit dem Button Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.16 dargestellt ist, erscheint wieder.
  – Will man Funktionswerte berechnen, so muß wie folgt vorgegangen werden:

    * Positioniere den Mauszeiger auf das entsprechende Eingabefeld (für jedes Argument der Funktion wird eine Eingabebox bereitgestellt).
    * Den dort angegebenen Wert auf die gewünschte Zahl korrigieren.

  – Nachdem alle Funktionsargumente eingegeben worden sind, kann der Funktionswert durch Anklicken des Buttons Berechne Funktionswert erzeugt werden.

    Konnte der Funktionswert berechnet werden, so wird dieser in der Funktionsergebnisbox angezeigt.

    Akzeptiert ProSiNN ein Funktionsargument nicht als Eingabe, dann wird dort 0 (bzw. 0.0) verwendet und der dazugehörige Funktionswert wird dargestellt.

- Durch Auswahl des Buttons Funktion ableiten wird ein weiteres Popup–Fenster erzeugt, das z.B. die Gestalt von Bild 18.18 aufweist:

Abbildung 18.18: Das Popup–Fenster zum *Funktion ableiten*

  – Mit dem Button Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.16 dargestellt ist, erscheint wieder.
  – Will man eine Funktion ableiten, so muß folgendermaßen vorgegangen werden:
    * Zuerst muß die Nummer der Funktionsvariablen, nach der abgeleitet werden soll, eingegeben werden. Die einzelnen Funktionsvariablen werden dabei von 1 bis zur Anzahl der Funktionsvariablen durchnummeriert. Die Eingabe erfolgt wie nachfolgend beschrieben:
      · Den Mauszeiger auf das Eingabefeld für die Nummer der Ableitungsvariablen positionieren.
      · Die dort angegebene ganze Zahl auf den gewünschten Wert korrigieren.
    * Der Bezeichner, der die abgeleitete Funktion anschließend repräsentiert, muß folgendermaßen eingegeben werden:
      · Den Mauszeiger auf das Eingabefeld für den Namen der Ableitungsfunktion positionieren.
      · Mit der Tastatur einen gültigen (siehe Abschnitt 4.1) noch nicht definierten Bezeichner eingeben.
  – Nachdem alle Eingaben vorgenommen worden sind, kann die Ableitung durch Anklicken des Buttons Berechne die Ableitung generiert werden.

    Konnte die Funktion abgeleitet werden, so erscheint wieder die Auswahlbox, die in Bild 18.16 dargestellt ist.

    War die Erzeugung der Ableitung nicht möglich, so wird eine Fehlermeldung ausgegeben und der Benutzer kann seine Eingaben korrigieren.

- Durch Auswahl des Buttons $\boxed{\text{Funktion bearbeiten}}$ wird ein weiteres Popup–Fenster erzeugt, das z.B. die Gestalt von Abbildung 18.19 aufweist:

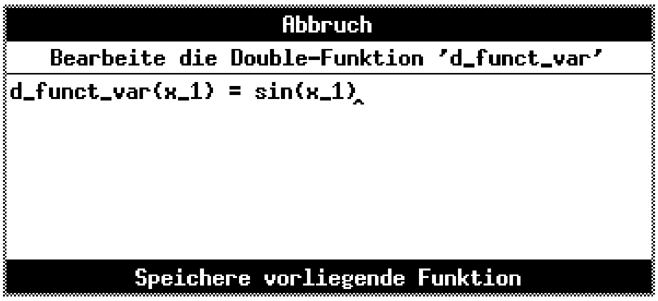

Abbildung 18.19: Das Popup–Fenster zum *Funktion bearbeiten*

  - Mit dem Button $\boxed{\text{Abbruch}}$ wird das Fenster beendet und die Auswahlbox, die in Bild 18.16 dargestellt ist, erscheint wieder.
  - Durch den Button $\boxed{\text{Speichere vorliegende Funktion}}$ wird der darüberliegende Text als Funktion des vorgegebenen Typs erzeugt:
    * Der eingegebene Funktionstext muß sich an folgende Syntax halten:

function_text	::=	**name***“(“* **var_list** *“)“* *“=“* **funct_expression.**
name	::=	**„Name der Funktionsvariablen“.**
var_list	::=	**var_name** { “,“ **var_name** }.
var_name	::=	**„Bezeichner (Funktionsvariable)“.**
funct_expression	::=	**term** { (“+“ | “–“) **term** }.
term	::=	**factor** { (“*“ | “/“) **factor** }.
factor	::=	**power** { „ˆ “ **power** }.
power	::=	[“–“] **scalar.**
scalar	::=	*“(“* **funct_expression** *“)“* | **function.**
function	::=	*“sin(“* **funct_expression** *“)“* |
		“cos(“ **funct_expression** *“)“* |
		“tan(“ **funct_expression** *“)“* |
		“asin(“ **funct_expression** *“)“* |
		“acos(“ **funct_expression** *“)“* |
		“atan(“ **funct_expression** *“)“* |
		“sinh(“ **funct_expression** *“)“* |
		“cosh(“ **funct_expression** *“)“* |
		“tanh(“ **funct_expression** *“)“* |
		“sqrt(“ **funct_expression** *“)“* |
		“ln(“ **funct_expression** *“)“* |
		“log(“ **funct_expression** *“)“* |
		var_or_const |
		own_function |
		var_name.
var_or_const	::=	**„Eine Konstante oder Variable“**
own_function	::=	**„Eine vom Benutzer definierte Funktion.“**

· **name** ist der Bezeichner, mit dem die Funktion im System gespeichert ist. Der Funktionsname kann auch geändert werden!

· **var_list** ist die durch Kommata getrennte Liste der Funktionsargumente. Diese Namen sind nur lokal bekannt. D.h. es können beliebige — auch vordefinierte — Bezeichner verwendet werden; die ursprüngliche Bedeutung wird kurzfristig verdeckt!

· Für **own_function** gilt die gleiche Syntax wie bei einer Funktionsauswertung; es dürfen aber zusätzlich noch alle vorkommenden Funktionsargumente verwendet werden!

· Die beiden Datentypen *int* und *double* können beliebig miteinander vermischt werden.

· Indem z.B. Operationen auf Konstanten zusammengefaßt werden, versucht ProSiNN **funct_expression** zu vereinfachen. Dabei werden aber Teilausdrücke nicht permutiert, so daß eine lokale und keine globale Vereinfachung stattfindet.

· Bei der Änderung einer Integerfunktion wird von allen vorkommenden Werten vom Typ *double* der Nachkommaanteil abgeschnitten. Analog gilt bei einer Funktionsauswertung innerhalb der Definition, daß alle Teilfunktionen vom Typ *double* und deren Werte intern auf den nächsten kleineren Integer abgebildet werden (bei negativen Zahlen werden sie auf den nächsten größeren Integer transformiert).

∗ Paßt der eingegebene Text nicht mehr in das in Abbildung 18.19 dargestellte Texteingabefenster, so erzeugt ProSiNN ein vertikales und/oder ein horizontales Scrollfeld. Mit ihnen kann der Anwender jeden Teil des Funktionstextes ändern.

Entdeckt das System einen Fehler im Eingabetext, so erscheint der invertierte Fehlerausgabebutton (mit der Fehlermeldung), der über dem Ein/Ausgabefenster plaziert wird (siehe Unterabschnitt 3.4.2). Nachdem der Benutzer den Button mit einer Maustaste angeklickt hat, ist die Fehlerstelle im Eingabefenster der Funktion invertiert dargestellt.

Wurde die Funktionsvariable korrekt geändert, so erscheint wieder die Auswahlbox, die in Bild 18.16 dargestellt ist.

- Durch Auswahl des Buttons [Funktion plotten] wird ein weiteres Popup–Fenster erzeugt, das z.B. die Gestalt von Abbildung 18.20 aufweist. Dieser Button ist nur dann in Abbildung 18.16 vorhanden, wenn die Funktion ein Argument hat und im System mindestens ein Grafikfenster existiert.

  – Mit dem Button [Abbruch] wird das Fenster beendet und die Auswahlbox, die in Bild 18.16 dargestellt ist, erscheint wieder.

  – Die Grafik, in die die Funktion ausgegeben werden soll, läßt sich folgendermaßen ändern:

    ∗ Positioniere den Mauszeiger auf das Eingabefeld für den Grafikkontext.

    ∗ Die dort angegebene ganze Zahl auf den gewünschten Grafikkontext korrigieren.

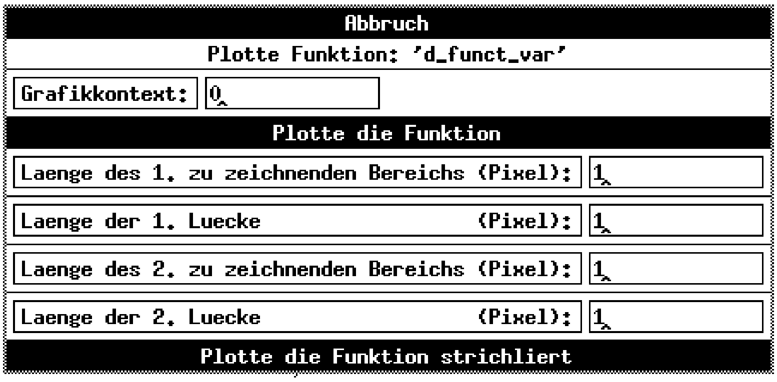

Abbildung 18.20: Das Popup–Fenster zum *Funktion plotten*

- Wird der Button Plotte die Funktion angewählt, so wird die Funktion über die gesamte Abszissenachse des Weltkoordinatensystems eingezeichnet.
- Soll die Funktion mit einer Strichlierung ausgegeben werden, muß wie folgt vorgegangen werden:
  * Ändere im Eingabefeld für die Länge des ersten zu zeichnenden Bereichs die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **block_1** in Abschnitt 8.4).
  * Ändere im Eingabefeld für die Länge der ersten Lücke die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **gap_1** in Abschnitt 8.4).
  * Ändere im Eingabefeld für die Länge des zweiten zu zeichnenden Bereichs die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **block_2** in Abschnitt 8.4).
  * Ändere im Eingabefeld für die Länge der zweiten Lücke die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **gap_2** in Abschnitt 8.4).

  Nachdem alle Eingaben vorgenommen worden sind, kann die Funktion durch Anklicken des Buttons Plotte die Funktion strichliert mit der angegebenen Strichlierung in das Grafikfenster eingezeichnet werden. Dabei wird die Funktion über die gesamte Abszissenachse des entsprechenden Weltkoordinatensystems ausgegeben.

Waren alle Eingaben korrekt und wurde die Funktion ausgegeben, so erscheint wieder die Auswahlbox, die in Abbildung 18.16 dargestellt ist.

Konnte die Funktion nicht gezeichnet werden, so wird dies durch eine Fehlermeldung gekennzeichnet und der Anwender kann seine Eingaben korrigieren.

18.4.5 „Benutzerkommandos bearbeiten“

Wenn der Benutzer mit dem Mauszeiger ein Benutzerkommando zum Bearbeiten ausgewählt hat, dann erscheint z.B. das Popup–Fenster, das in Bild 18.21 abgebildet ist (hier wurde das Makro `macro_var`, das in Abbildung 18.11 dargestellt ist, ausgewählt):

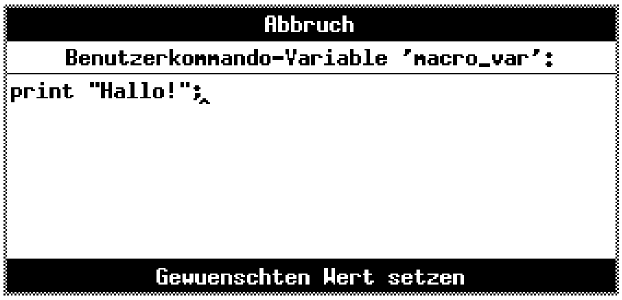

Abbildung 18.21: Das Popup–Fenster zum *Benutzerkommandos bearbeiten*

Bem.:

- Durch Anklicken des Buttons Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.11 dargestellt ist, erscheint wieder.
- Will der Anwender den Wert der dargestellten Variablen ändern, so muß er folgendermaßen vorgehen:
  - Den Mauszeiger auf das Eingabefeld für den Text setzen.
  - Der Kommandotext kann nun korrigiert werden. Dabei muß folgende Syntax eingehalten werden:

    macro_text ::= **command** *“;“* { **command** *“;“* }.
    command ::= **„Ein gültiger ProSiNN–Befehl“.**

    * **command** bezeichnet ein beliebiges von ProSiNN unterstütztes Kommando. Auch der Name eines anderen bereits generierten Makros ist zulässig.
    * Die Korrektheit der Anweisungen wird erst beim ersten Aufruf des Makros überprüft!
    * Alle im Anweisungsblock neu erzeugten Variablen sind auch nach Ausführung des Makros noch definiert. Es können also keine lokalen Variablen generiert werden.
  - Paßt der eingegebene Text nicht mehr in das in Bild 18.21 dargestellte Eingabefenster, so erzeugt ProSiNN ein horizontales und/oder ein vertikales Scrollfeld. Mit ihnen kann jeder Teil des Makrotextes erreicht werden.
- Nachdem der Benutzer die Modifikationen vorgenommen hat, kann der Kommandotext durch Anklicken des Buttons Gewuenschten Wert setzen an die Variable zugewiesen werden.

  Anschließend wird das Fenster durch den Abbruch–Button wieder verlassen.

18.4.6 „Netzvariablen bearbeiten“

Wenn der Anwender mit dem Mauszeiger eine Netzvariable zum Bearbeiten ausgewählt hat, dann erscheint z.B. das Popup–Fenster, das in Abbildung 18.22 dargestellt ist (hier wurde die Gewichtsvariable `weight_var`, die in Bild 18.11 abgebildet ist, ausgewählt):

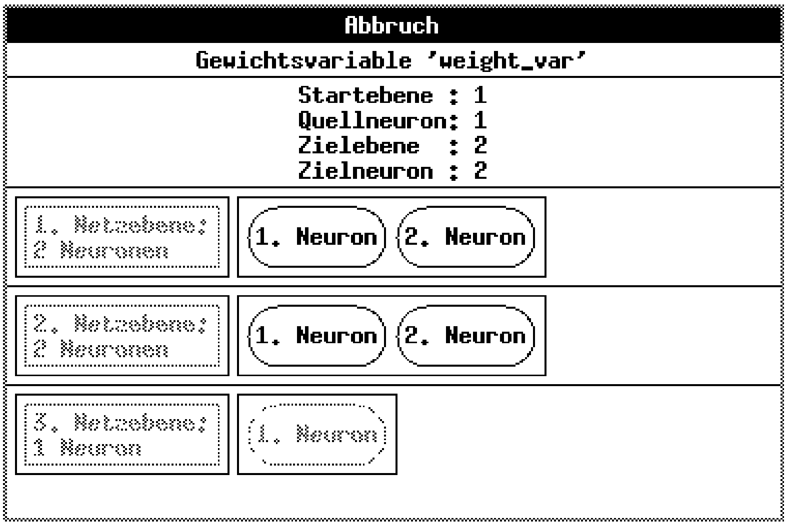

Abbildung 18.22: Das Popup–Fenster zum *Netzvariablen bearbeiten*

Bem.:

- Durch Anklicken des Buttons $\boxed{\text{Abbruch}}$ wird das Fenster beendet und die Auswahlbox, die in Bild 18.11 dargestellt ist, erscheint wieder.
- Ist das Netz zu groß für das in Abbildung 18.22 dargestellte Fenster, so wird von ProSiNN ein horizontales und ein vertikales Scrollfeld erzeugt. Mit ihnen kann der Benutzer jede Strukturkomponente erreichen.
- Soll eine Variable geändert werden, so muß wie folgt vorgegangen werden:
  - Liegt eine Variable vom Typ *layer* vor, dann wird mit dem Anklicken der gewünschten Ebene der Inhalt der Variablen entsprechend geändert.
  - Handelt es sich um eine Variable vom Typ *unit*, dann wird durch das Anklicken der gewünschten Zelle die Variable neu gesetzt.
  - Liegt eine Variable vom Typ *weight* vor, dann wird mit dem ersten Anklicken einer Zelle das Startneuron des Gewichtes gesetzt. Anschließend sind alle Zellen inaktiv, die nicht in der nachfolgenden Ebene liegen. Nun muß mit der Maus das Zielneuron gesetzt werden. Nach diesem Schritt wird der Inhalt der Variablen geändert.

  Der aktuelle Inhalt der Variablen wird über der Netzstruktur angezeigt. Wird eine Variable geändert, dann wird auch der Informationstext aktualisiert!

18.5 Der Menüpunkt „Grafikfenster bearbeiten“

Zur Bearbeitung von Grafikfenstern kann der Menüpunkt *Grafikfenster bearbeiten* angewählt werden. Anschließend erscheint das Popup–Fenster, das in Bild 18.23 dargestellt ist (diese Form erhält man, wenn im System keine Grafikfenster existieren):

Abbildung 18.23: Die Auswahlbox des Menüpunktes *Grafikfenster bearbeiten*

Bem.:

- Mit dem Button Abbruch kann der Menüpunkt beendet werden.
- Durch Anklicken des Buttons Neues Grafikfenster erzeugen kann ein Grafikfenster generiert werden. Dazu wird ein weiteres Popup–Fenster, das in Abbildung 18.24 dargestellt ist, erzeugt:

Abbildung 18.24: Das Popup–Fenster zum Erzeugen eines Grafikfensters

  - Mit dem Button Abbruch wird das Fenster ohne Generierung einer neuen Grafik beendet und das Popup–Fenster, das in Bild 18.23 dargestellt ist, erscheint wieder.
  - Zur Änderung der Grafikfenstergröße muß folgendermaßen vorgegangen werden:
    * Den Mauszeiger auf das Eingabefeld für die Höhe oder die Breite setzen.
    * Die dort angegebene ganze Zahl auf den gewünschten Wert korrigieren. Dabei ist folgendes zu beachten:
      · Die Breite des Grafikfensters muß im Intervall $[1, \ldots, 2000]$[2] liegen. Die Mindestbreite, die ProSiNN immer erzeugt, ist 100 Pixel.
      · Die Höhe des Grafikfensters muß im Intervall $[1, \ldots, 2000]$[3] liegen. Die Mindesthöhe, die ProSiNN immer erzeugt, ist 100 Pixel.

[2] Der Wert für die maximale Breite eines Fensters kann bei der Systeminstallation geändert werden
[3] Der Wert für die maximale Höhe eines Fensters kann bei der Systeminstallation geändert werden

- Nachdem die gewünschten Geometrievorgaben gesetzt wurden, kann das Fenster durch Anwählen des Buttons Erzeuge neues Grafikfenster generiert werden.

  Waren alle Eingaben korrekt, dann muß mit der Maus das neue Fenster positioniert werden. Solange dies nicht ausgeführt wird, wartet das System. Anschließend erscheint wieder das Fenster, das in Abbildung 18.23 dargestellt ist.

  Lagen falsche Geometrieangaben vor, dann wird eine Fehlermeldung ausgegeben und der Anwender kann seine Eingaben korrigieren.
- Ein Fenster kann nachträglich verkleinert oder vergrößert werden. Intern bleiben aber immer die maximalen Abmesungen erhalten; d.h. wenn der Benutzer z.B. ein Fenster nach dessen Erzeugung erst vergrößert und anschließend wieder verkleinert, bleiben dessen maximale Abmessungen, die nach der Vergrößerung gültig waren, erhalten. Bei der Verkleinerung eines Fensters wird immer das Zentrum des intern vorliegenden Gesamtbildes dargestellt.

- Durch den Button Alle Grafikfenster loeschen werden alle Grafikfenster eliminiert. Dabei gehen deren Inhalte verloren. Um eine erzeugte Grafik zu speichern, muß der Anwender auf *X Window Utilities* zurückgreifen.
- Wenn im System mindestens ein Grafikfenster existiert, dann hat die Auswahlbox des Menüpunktes *Grafikfenster bearbeiten* z.B. die in Bild 18.25 gezeigte Gestalt:

Abbildung 18.25: Die Auswahlbox des Menüpunktes *Grafikfenster bearbeiten*

Um nun *ein* Grafikfenster zur Bearbeitung auszuwählen, muß mit der Maus die Zahl, die den Grafikkontext repräsentiert, angeklickt werden. Nach der Selektion einer Grafik erscheint z.B. das Popup–Fenster, das in Abbildung 18.26 dargestellt ist (hier wurde das Fenster mit dem Kontext `0` gewählt).

- Mit dem Button Abbruch wird dieses Fenster beendet und das Popup–Fenster, das in Bild 18.25 dargestellt ist, erscheint wieder.
- Unterhalb des Abbruch–Buttons wird von dem selektierten Fenster der Grafikkontext und das Weltkoordinatensystem angezeigt.
- Durch Anklicken eines der übrigen Buttons wird in der Regel ein weiteres Fenster erzeugt, mit dem die gewünschte Aktion durchgeführt werden kann. Die einzelnen Auswahlpunkte werden in den folgenden Unterabschnitten beschrieben.

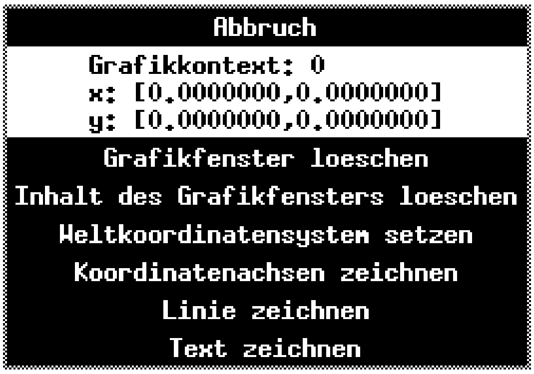

Abbildung 18.26: Die Auswahlbox zum Bearbeiten eines Grafikfensters

18.5.1 „Grafikfenster loeschen“

Durch Anklicken des Buttons Grafikfenster loeschen wird das selektierte Fenster aus dem System entfernt und die Auswahlbox, die in Abbildung 18.25 dargestellt ist, erscheint wieder.

Bem.:

Der Inhalt des eliminierten Fensters geht verloren. Um eine erzeugte Grafik zu speichern, muß der Anwender auf *X Window Utilities* zurückgreifen.

18.5.2 „Inhalt des Grafikfensters loeschen“

Durch Auswahl des Buttons Inhalt des Grafikfensters loeschen wird der Inhalt das ausgewählten Fensters gelöscht.

Bem.:

- Das gesetzte Weltkoordinatensystem bleibt erhalten!
- Um eine erzeugte Grafik zu speichern, muß der Benutzer auf *X Window Utilities* zurückgreifen.

18.5.3 „Weltkoordinatensystem setzen“

Durch Anklicken des Buttons Weltkoordinatensystem setzen wird für das selektierte Fenster ein Popup–Fenster, das in Bild 18.27 dargestellt ist, erzeugt.

Bem.:

- Mit dem Button Abbruch wird das Fenster ohne Modifikation des Weltkoordinatensystems beendet und die Auswahlbox, die in Abbildung 18.26 dargestellt ist, erscheint wieder.
- Zur Eingabe des Weltkoordinatensystems muß wie folgt vorgegangen werden:

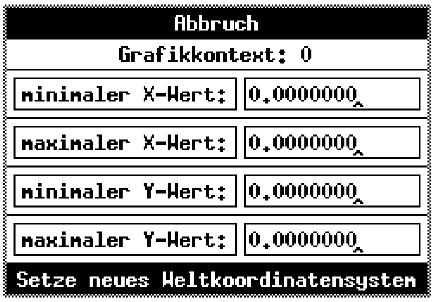

Abbildung 18.27: Das Popup–Fenster zum *Weltkoordinatensystem setzen*

- Den Mauszeiger auf das zu ändernde Eingabefeld setzen.
- Die angegebene Fließkommazahl auf den gewünschten Wert korrigieren. Dabei ist folgendes zu beachten:
  - Der minimale `X`–Wert muß kleiner als der maximale `X`–Wert sein. Sie repräsentieren den linken und rechten Rand des Weltkoordinatensystems.
  - Der minimale `Y`–Wert muß kleiner als der maximale `Y`–Wert sein. Sie stellen den unteren und oberen Rand des Weltkoordinatensystems dar.

- Nachdem der Anwender die Bereichsvorgaben eingegeben hat, kann das Weltkoordinatensystem durch Anwählen des Buttons Setze neues Weltkoordinatensystem gültig gemacht werden.

  Konnte das Weltkoordinatensystem gesetzt werden, so erscheint wieder die Auswahlbox, die in Bild 18.26 dargestellt ist.

  Ist eine Bereichsangabe unkorrekt, dann wird eine Fehlermeldung ausgegeben und der Benutzer kann seine Eingaben korrigieren.

- Das System transformiert das Weltkoordinatensystem so in die Grafik, daß an jedem Rand noch 20 Pixel Leerraum verbleibt.

18.5.4 „Koordinatenachsen ausgeben“

Durch Anklicken des Buttons Koordinatenachsen zeichnen wird für das ausgewählte Fenster ein Popup–Fenster, das in Abbildung 18.28 dargestellt ist, erzeugt (die hier vorliegenden Bereiche des Weltkoordinatensystems sind `X`: $[-10.0, \ldots, 10.0]$ und `Y`: $[-1.0, \ldots, 1.0]$).

Bem.:

- Mit dem Button Abbruch wird das Fenster beendet und die Auswahlbox, die in Bild 18.26 dargestellt ist, erscheint wieder.

- Zur Eingabe der Koordinatenachsenbereiche, muß man wie folgt vorgehen:
  - Den Mauszeiger auf das zu ändernde Eingabefeld setzen.
  - Die dort angegebene Fließkommazahl auf den gewünschten Wert korrigieren. Dabei ist folgendes zu beachten:

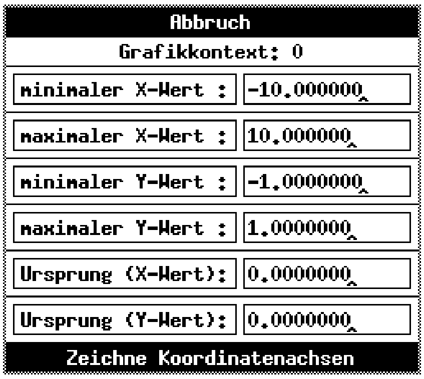

Abbildung 18.28: Das Popup–Fenster zum *Koordinatenachsen zeichnen*

  * Der minimale `X`–Wert muß kleiner als der maximale `X`–Wert sein. Sie repräsentieren den linken und rechten Rand der Koordinatenachsen.
  * Der minimale `Y`–Wert muß kleiner als der maximale `Y`–Wert sein. Sie stellen den unteren und oberen Rand der Koordinatenachsen dar.
  * Die Ursprungswerte `X` und `Y` bilden die Position des Koordinatenachsenursprungs. Diese Werte werden in der Nähe des Ursprungs in Klammern eingezeichnet. Sie dürfen beliebig gesetzt werden; verlassen sie aber den Bereich, der durch die darüberliegenden Koordinatenwerte gesetzt wurde, wird der Ursprung an den entsprechenden Rand verschoben.
  * Liegt der Bereich, der durch die minimalen und maximalen `X/Y`–Werte angegeben wurde, nicht im Weltkoordinatensystem, so ist das Achsenkreuz nicht sichtbar.

- Nachdem der Anwender die Bereichsvorgaben eingegeben hat, können die Koordinatenachsen durch Anwählen des Buttons [Zeichne Koordinatenachsen] in die Grafik eingezeichnet werden.

  Sind die Koordinatenachsen ausgegeben worden, dann erscheint wieder die Auswahlbox, die in Abbildung 18.26 dargestellt ist.

  Ist eine Bereichsangabe unkorrekt, so wird eine Fehlermeldung ausgegeben und der Benutzer kann seine Eingaben korrigieren.

18.5.5 „Linie ausgeben“

Durch Anklicken des Buttons [Linie zeichnen] wird für das selektierte Fenster ein Popup–Fenster, das in Bild 18.29 dargestellt ist, erzeugt (die hier vorliegenden Bereiche des Weltkoordinatensystems sind `X`: $[-10.0, \ldots, 10.0]$ und `Y`: $[-1.0, \ldots, 1.0]$).

Bem.:

- Mit dem Button [Abbruch] wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.26 dargestellt ist, erscheint wieder.

- Zur Eingabe der Anfangs– und Endpunkte der Linie, muß man wie folgt vorgehen:

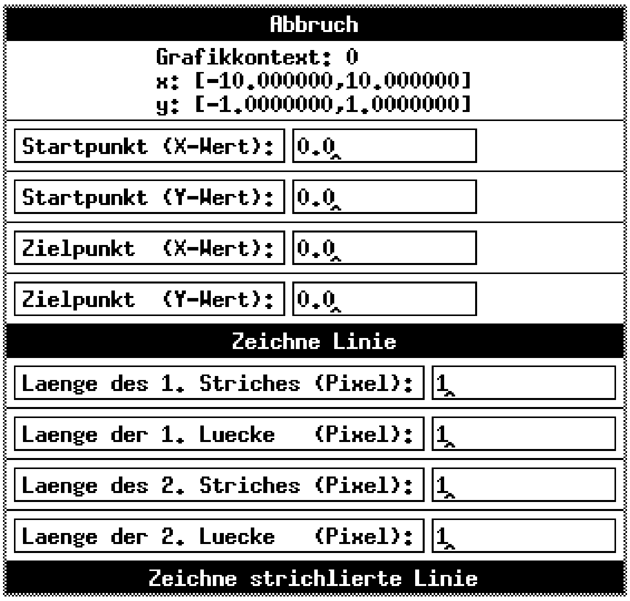

Abbildung 18.29: Das Popup–Fenster zum *Linie zeichnen*

  – Den Mauszeiger auf das zu ändernde Eingabefeld setzen.
  – Die dort angegebene Fließkommazahl auf den gewünschten Wert korrigieren (alle Angaben sind in Weltkoordinaten einzugeben).

- Nachdem der Anwender die Start– und Zielpunktvorgaben eingegeben hat, kann die Linie durch Anwählen des Buttons Zeichne Linie in die Grafik eingezeichnt werden.

  Wenn die Linie ausgegeben wurde, dann erscheint wieder die Auswahlbox, die in Bild 18.26 dargestellt ist.

  Ist ein Anfangs– oder Endpunkt unkorrekt, so wird eine Fehlermeldung ausgegeben und der Benutzer kann seine Eingaben korrigieren.

- Soll die Linie mit einer Strichlierung (siehe Abschnitt 8.4) ausgegeben werden, dann muß folgendermaßen vorgegangen werden:

  – Ändere im Eingabefeld für die Länge des ersten zu zeichnenden Striches die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **block_1** in Abschnitt 8.4).
  – Ändere im Eingabefeld für die Länge der ersten Lücke die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **gap_1** in Abschnitt 8.4).
  – Ändere im Eingabefeld für die Länge des zweiten zu zeichnenden Striches die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **block_2** in Abschnitt 8.4).
  – Ändere im Eingabefeld für die Länge der zweiten Lücke die dort angegebene ganze Zahl auf den gewünschten Wert (dies entspricht dem Nichtterminalsymbol **gap_2** in Abschnitt 8.4).

Sind alle Eingaben vorgenommen worden, dann kann die Gerade durch Anklicken des Buttons Zeichne strichlierte Linie mit der angegebenen Strichlierung in das Grafikfenster eingezeichnet werden.

Wurde die Linie ausgegeben, so erscheint wieder die Auswahlbox, die in Abbildung 18.26 dargestellt ist.

Ist eine Angabe unkorrekt, dann wird eine Fehlermeldung ausgegeben und der Anwender kann seine Eingaben korrigieren.

18.5.6 „Text ausgeben“

Durch Anklicken des Buttons Text zeichnen wird für das ausgewählte Fenster ein Popup–Fenster, das in Bild 18.30 dargestellt ist, erzeugt (die hier vorliegenden Bereiche des Weltkoordinatensystems sind `X`: $[-10.0, \ldots, 10.0]$ und `Y`: $[-1.0, \ldots, 1.0]$):

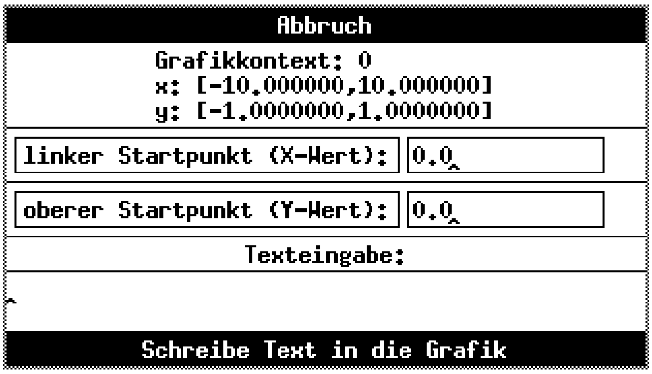

Abbildung 18.30: Das Popup–Fenster zum *Text zeichnen*

Bem.:

- Mit dem Button Abbruch wird das Fenster beendet und die Auswahlbox, die in Abbildung 18.26 dargestellt ist, erscheint wieder.
- Zur Eingabe der linken oberen Textposition muß wie folgt vorgegangen werden:
  - Den Mauszeiger auf das zu ändernde Eingabefeld setzen.
  - Die dort angegebene Fließkommazahl auf den gewünschten Startwert korrigieren. Dabei ist folgendes zu beachten:
    - ∗ Der linke Startpunkt setzt den linken Rand des Textes in Weltkoordinaten.
    - ∗ Der obere Startpunkt spezifiziert den oberen Rand des Textes in Weltkoordinaten.
- Der Text, der ausgegeben werden soll, muß in das Feld eingegeben werden, das sich über dem Button Schreibe Text in die Grafik befindet. Dabei ist zu beachten:
  - Die Zeichenkette kann nur aus einer Zeile bestehen (die `Return`–Taste wird ignoriert).

  - Ein leerer Text wird nicht ausgegeben.
  - Paßt der Text nicht mehr in den für den Anwender sichtbaren Bereich, dann erzeugt ProSiNN zum Erreichen jedes Zeichens ein horizontales Scrollfeld.

- Nachdem der Benutzer den Text und die Startposition eingegeben hat, kann die Zeichenkette durch Anklicken des Buttons [Schreibe Text in die Grafik] ausgegeben werden.

  Wurde der Text eingezeichnet, so erscheint wieder die Auswahlbox, die in Bild 18.26 dargestellt ist.

  Ist eine Angabe unkorrekt, dann wird eine Fehlermeldung ausgegeben und der Anwender kann seine Eingaben korrigieren.

18.6 Der Menüpunkt „Alle Variablen freigeben“

Will der Benutzer *bestimmte* Variablen und Strukturen wieder freigeben, so muß er den Menüpunkt *Alle Variablen freigeben* anwählen. Anschließen erscheint die Auswahlbox, die in Abbildung 18.31 dargestellt ist:

Abbildung 18.31: Das Popup–Fenster des Menüpunktes *Alle Variablen freigeben*

Bem.:

- Durch Anklicken des Buttons [Abbruch] wird der Menüpunkt abgebrochen.
- Durch den Button [Alle Variablen freigeben] wird folgendes durchgeführt:
  - Alle mit der *define*–Anweisung erzeugten Variablen werden wieder freigegeben. Dazu gehören Integer–, Double–, Feld–, Funktions–, String–, Layer–, Unit– und Weight–Variablen.
  - Alle durch *alias* generierten Makros werden aus dem System entfernt.
  - Ein Neuronales Netz bleibt im System enthalten!
  - Testmuster werden gelöscht.
  - Trainingsmuster bleiben erhalten; deren Bezeichner — sofern sie existieren — werden aber entfernt!
  - Die Aktivierungsfunktion bleibt unverändert!
  - Alle Dateien und Grafikfenster werden geschlossen.
  - Alle Text– und Grafikprotokolle werden gelöscht.

  Nach Ausführung dieses Menüpunktes existieren *keine* vom Anwender definierten Bezeichner mehr im System (die Ausnahme bilden die beiden Bezeichner der Aktivierungsfunktion und deren Ableitung — sofern der Benutzer eine neue Aktivierungsfunktion definiert hatte)!

Kapitel 19

Die Menüs „ProSiNN Info" und „Programm beenden"

Dieses Kapitel beschäftigt sich mit dem Menübutton *ProSiNN Info*, der Informationen über das System und den Autor liefert, und dem Menüpunkt *Programm beenden*, der das System beendet.

19.1 Der Menübutton „ProSiNN Info"

Um Informationen über ProSiNN und dessen Autor zu erhalten, muß der Benutzer den Menübutton *ProSiNN Info* anklicken. Anschließend wird ein Popup–Fenster über dem Zentrum des Ein/Ausgabefensters generiert:

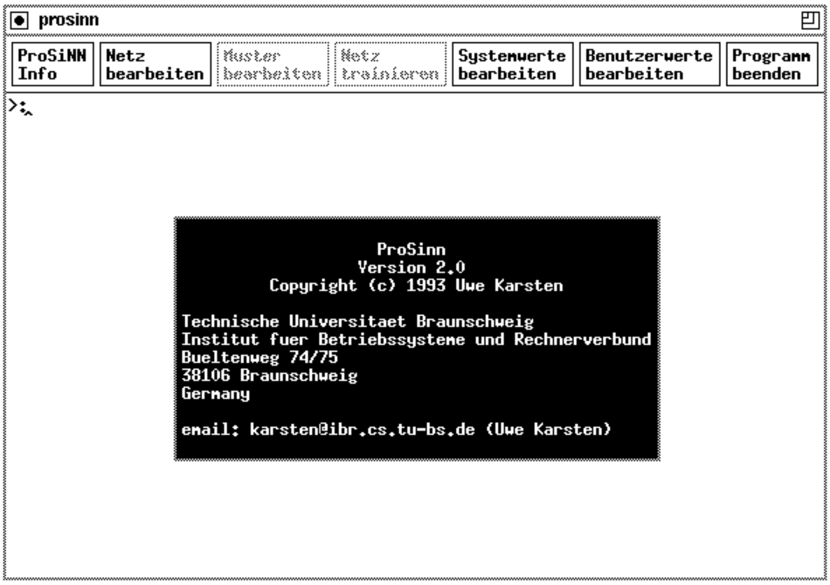

Abbildung 19.1: Das Popup–Fenster zum Menübutton *ProSiNN Info*

Soll mit dem System weitergearbeitet werden, dann muß mit der linken Maustaste das Informations–Fenster angewählt werden.

19.2 Der Menüpunkt „Programm beenden“

Will der Anwender das Programmsystem ProSiNN beenden, so muß er im Menü *Programm beenden* (siehe Abbildung 19.2) den Menüpunkt *System beenden* anwählen.

Abbildung 19.2: Das Menü *Programm beenden*

Teil IV

Anwendungsbeispiele

Kapitel 20

Das Training von Neuronalen Netzen mit ProSiNN

In diesem Kapitel werden Anwendungsbeispiele vorgeführt, die die folgenden drei Probleme und deren Lösbarkeit mit dem ProSiNN–Programmsystem untersuchen:

- Das XOR–Problem
- Das Encoder–Problem
- Das T–C Problem

Diese Aufgaben wurden auch von Rumelhart, Hinton und Williams (siehe [RUM86]) bei der praktischen Erprobung der Lernregel für das mehrschichtige Perceptron verwendet.

Bem.:

Bei der Problemlösung wird — sofern nicht anders angegeben ist — immer von dem Systemzustand ausgegangen, der beim Start von ProSiNN vorliegt. D.h. es gilt folgendes:

- Es wird die sigmoide Aktivierungsfunktion benutzt.
- Der quadratische Fehlermaßmodus ist gesetzt.
- Die untere Fehlerschranke beträgt 0.001.
- Die Lernrate ist 0.2 und als Lernmoment wird 0.9 verwendet.
- Der Mißerfolgsfaktor ist 1.04.
- Der Erfolgslernratenfaktor beträgt 1.05 und der Mißerfolgslernratenfaktor ist 0.7.

20.1 Das XOR–Problem

Eine klassische Aufgabenstellung, bei der innere Schichten benötigt werden, ist das XOR–Problem (siehe auch Unterabschnitt 2.2.2). In Tabelle 20.1 wird die Zuordnung der beiden binären Eingabewerte zu der binären Netzantwort dargestellt:

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Tabelle 20.1: Das XOR–Problem

Die Schwierigkeit besteht hier darin, daß ein Netz linear nicht separable Muster erlernen soll!

Aufgabe:

Die in Tabelle 20.1 angegebene Abbildung soll ein Neuronales Netz erlernen. Dabei ist folgendes zu beachten:

- Die innere Schicht enthält zwei Neuronen.
- Es soll nur das Menüsystem verwendet werden.
- Ein grafisches Protokoll, das das aktuelle Fehlermaß ausgibt, ist zu definieren.

Lösung:

In der nachfolgenden Aufzählung wird beschrieben, wie die Aufgabenstellung gelöst wird.

- Zuerst wird die Datei, die die Trainingsmuster enthält, generiert.
  - In dem Menü *Benutzerwerte bearbeiten* wird der Menüpunkt *Datei editieren* mit dem Mauszeiger angeklickt.
  - In der anschließend erscheinenden Dateiauswahlbox wird `xor.pat` als Name für die zu erzeugende Trainingsmusterdatei eingegeben.
  - Danach wird mit dem Button $\boxed{\text{Datei editieren}}$ das Editorfenster für die gewünschte Datei eröffnet.
  - In der in Bild 20.1 dargestellten Situation wurden bereits alle benötigten Eingaben vorgenommen.
  - Mit dem Button $\boxed{\text{Datei speichern}}$ wird das File unter dem angegebenen Namen gespeichert.
  - Anschließend wird der Button $\boxed{\text{Abbruch}}$ ausgewählt. Das Editorfenster wird nun in dem erzeugten Auswahl–Fenster durch den Button $\boxed{\text{Editor verlassen}}$ beendet.

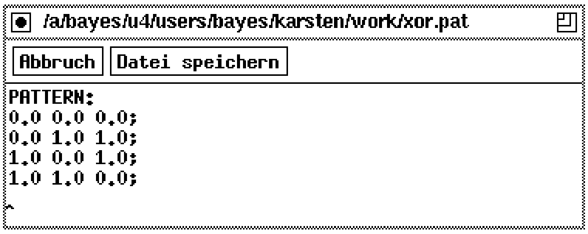

Abbildung 20.1: Das Editorfenster der XOR–Trainingsmusterdatei

- Jetzt ist die Netzstruktur zu erzeugen.
  - In dem Menü *Netz bearbeiten* wird der Menüpunkt *Netz erzeugen* ausgewählt.
  - Nachdem alle benötigten Eingaben vorgenommen worden sind, hat das Popup–Fenster den in Abbildung 20.2 dargestellten Inhalt:

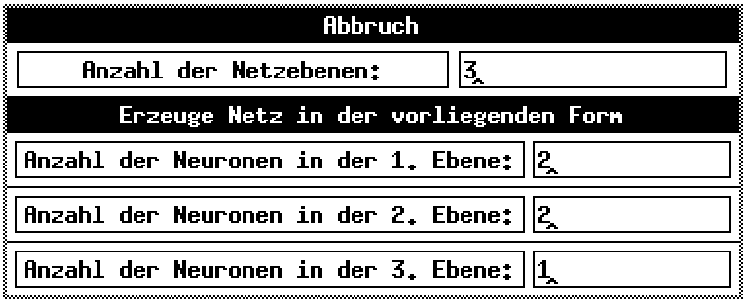

Abbildung 20.2: Das Popup–Fenster des Menüpunktes *Netz erzeugen*

  - Mit dem Button Erzeuge Netz in der vorliegenden Form wird das Netz generiert.

- Nun muß das Netz initialisiert werden.
  - In dem Menü *Netz bearbeiten* wird der Menüpunkt *Netz initialisieren* ausgewählt.
  - Mit dem Button Initialisiere das Netz mit den gegebenen Vorgaben wird das Netz initialisiert und das erzeugte Popup–Fenster wieder beendet.

- Die Trainingsmuster werden jetzt geladen.
  - In dem Menü *Muster bearbeiten* wird der Menüpunkt *Trainingsmuster laden* angeklickt.
  - In der anschließend erscheinenden Dateiauswahlbox wird die Datei `xor.pat` als Trainingsmusterfile gewählt.
  - Die Muster werden danach mit dem Button Trainingsmuster laden eingelesen.

- Die Grafik, in die protokolliert werden soll, ist nun zu generieren.
  - In dem Menü *Benutzerwerte bearbeiten* wird der Menüpunkt *Grafikfenster bearbeiten* angeklickt.

- In dem erscheinenden Popup–Fenster (siehe Bild 18.23) wird der Auswahlbutton Neues Grafikfenster erzeugen angewählt.
- Das nun generierte Popup–Fenster (siehe Abbildung 18.24) wird mit dem Button Erzeuge neues Grafikfenster verlassen. Parallel dazu wird das Grafikfenster erzeugt.
- In dem nun vorliegenden Popup–Fenster (siehe z.B. Bild 18.25) muß der Grafikkontext 0 angeklickt werden. Anschließend erscheint das Fenster, das in Abbildung 18.26 dargestellt ist.
- Mit dem Button Weltkoordinatensystem setzen wird ein weiteres Popup–Fenster generiert.
- Das in Bild 20.3 dargestellte Fenster hat den Inhalt, der nach den hier verwendeten Eingaben vorliegt (die hier benutzten Bereiche des Weltkoordinatensystems sind `X`: $[0.0, \ldots, 2000.0]$ und `Y`: $[0.0, \ldots, 1.0]$):

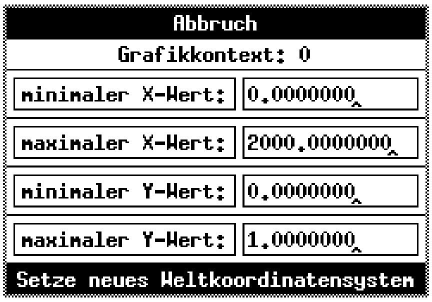

Abbildung 20.3: Das Fenster zum Setzen des Weltkoordinatensystems

- Mit dem Button Setze neues Weltkoordinatensystem wird das Weltkoordinatensystem generiert und das Fenster, das in Abbildung 18.26 dargestellt ist, erscheint wieder.
- Mit dem Button Koordinatenachsen zeichnen wird ein weiteres Popup–Fenster erzeugt.
- Nachdem alle Eingaben vorgenommen worden sind, hat das Fenster den in Bild 20.4 dargestellten Inhalt.
- Mit dem Button Zeichne Koordinatenachsen werden die Koordinatenachsen in die Grafik eingezeichnet und das Fenster, das in Abbildung 18.26 dargestellt ist, erscheint wieder.
- Durch (zweimaliges) Anwählen des jeweils vorhandenen Abbruch–Buttons wird dieser Menüpunkt beendet.

• Nun muß noch das Grafikprotokoll definiert werden.

  - In dem Menü *Netz trainieren* wird der Menüpunkt *Bearbeite Grafikprotokoll* ausgewählt.
  - Nachdem die verlangten Eingaben vorgenommen worden sind, hat das generierte Popup–Fenster den in Bild 20.5 dargestellten Inhalt: