Mitteilungen aus der Biologischen Bundesanstalt
für Land- und Forstwirtschaft
Berlin-Dahlem

Evaluation and improvement of mathematical models of pesticide mobility in soils and assessment of their potential to predict contamination of water systems

by

Dr. Allan Walker (Scientific coordinator)
Horticulture Research International
Wellesbourne, England
Prof. R. Calvet
Institute National Agronomique
Thiverval-Grignon, France
Prof. A.A.M. Del Re
Universita Cattolica del Sacro Cuore
Piacenza, Italia
Prof. Dr. W. Pestemer
Biologische Bundesanstalt für Land- und Forstwirtschaft,
Berlin (Dahlem) Deutschland
J.M. Hollis
Soil Survey and Land Research Centre (Cranfield University)
Silsoe England

Heft 307

Berlin 1995

Herausgegeben
von der Biologischen Bundesanstalt für Land- und Forstwirtschaft
Berlin-Dahlem

Blackwell Wissenschafts-Verlag GmbH Berlin/Wien
Kurfürstendamm 57, D-10707 Berlin

ISSN 0067-5849 ISBN 3-8263-3069-2

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Evaluation and improvement of mathematical models of pesticide mobility in soils and assessment of their potential to predict contamination of water systems
Hrsg. von der Biologischen Bundesanstalt für Land- und Forstwirtschaft Berlin-Dahlem. By. Allan Walker ... – Berlin; Wien: Blackwell-Wiss.-Verl. [in Komm.], 1995.
(Mitteilungen aus der Biologischen Bundesanstalt für Land- und Forstwirtschaft Berlin-Dahlem; H. 307)
ISBN 3-8263-3069-2
NE: Walker, Allan; Biologische Bundesanstalt für Land- und Forstwirtschaft <Berlin; Braunschweig>:
Mitteilungen aus der...

1995 Kommissionsverlag Blackwell Wissenschafts-Verlag GmbH Berlin/Wien, Kurfürstendamm 57, 10707 Berlin
Printed in Germany by Arno Brynda, Berlin

Evaluation and improvement of mathematical models of pesticide mobility in soils and assessment of their potential to predict contamination of water systems

Final Report of a Research Project of the EU

in the ENVIRONMENT RESEARCH PROGRAMME

(Research Area IV.2.1)

'Risk from Agricultural Technologies'

(Contract no.: EV5V-CT92-0226)

Author's addresses, project leaders and support scientists:

A. **WALKER, A.,** (Scientific coordinator)

Melacini, A.
Cullington, J.E.

Crop Protection Division
Horticulture Research International
Wellesbourne
UK-WARWICK CV35 9EF

B. CALVET, R.

Baer, Ulrike
Dabadie, J.-M.

Unité de Science du Sol
Institute National Agronomique
Centre de Grignon
F-78850 THIVERVAL-GRIGNON

C. DEL RE, A.A.M.

Trevisan, M.
Capri, E.

Istituto Chimica Agraria e Ambientale
Facoltà di Agraria Piacenza
Università Cattolica del Sacro Cuore
Via Emilia Parmense 84
I-29100 PIACENZA

D. PESTEMER, W.

Günther, Petra

Federal Biological Research Centre for
Agriculture and Forestry
Institute for Ecological Chemistry
Königin-Luise-Strasse 19
D-14195 BERLIN (DAHLEM)

E. HOLLIS, J.M.

Brown, C.D.
Siddons, Penny

Soil Survey and Land Research Centre
School of Agriculture, Food and the Environment
Cranfield University
UK-SILSOE MK45 4DT

Contents

1. INTRODUCTION

Intensive crop production systems throughout Europe almost all rely heavily on inputs of synthetic pesticides for weed, disease and pest control. There is increasing awareness that extensive pesticide use, particularly of residual soil-applied herbicides, can lead to contamination of surface and ground water. The movement of pesticides in soils is controlled by complex interactions between a number of processes including the dose, frequency and timing of application, adsorption-desorption phenomena, rates of degradation in the soil, other dissipation routes such as volatilization, rates and routes of water flow, and hydrological features such as depth to aquifers, and run-off characteristics.

Field and laboratory experiments to investigate behaviour can only cover a small number of the different situations in which any pesticide is likely to be used in practice. One approach to estimating the likely variability in behaviour is computer modelling, and several models of pesticide behaviour are now available. They operate with different levels of complexity and with different input data requirements. Some have been developed to evaluate a single component of the dissipation process such as degradation, volatilization or movement in soil; others describe interactions between several processes. None is ideal in all respects and it may be that a combination of different models or model components may be required to fully evaluate environmental fate of a specific chemical. The main objectives of this project were:

1. To compile a data base of the results from recent studies of pesticide redistribution in soils and of water contamination in selected areas of Germany, France, Italy, the United Kingdom, and other European Countries.
2. To bring together a range of pesticide fate and behaviour models, to itemise their data input requirements, and to evaluate their predictive ability when tested against data sets of different levels of complexity and from diverse agricultural areas.
3. To carry out some parameter sensitivity analyses and to investigate use of the models for predicting variability in leaching behaviour.
4. To make a modular analysis of selected models so that programming structures

might be developed to permit selection of model sub-routines most appropriate to specific situations.

5. To develop an improved, integrated modelling system that can be used for research or advisory purposes.

A number of papers describing results from this research project have already been published, and it is the intention that as much of the work as possible will be made available to the wider scientific community in this way. The results presented in this report represent just a small cross section of those available, and they have been selected to illustrate the overall data types that have been obtained. More detailed results can be found in the following publications:

Barriuso, E., Calvet, R., Baer, U and Dabadie, J.M. (1994) Field atrazine behaviour and dissipation kinetics in two different soils. In: *Environmental Behaviour of Pesticides and Regulatory Aspects, Proceedings COST66 Symposium, Brussels*, 390-395.

Del Re, A.A.M., Trevisan, M., Capri, E., Evans, S.P. and Brasa, E (1994) Criteria of existing model evaluation. In: *Environmental Behaviour of Pesticides and Regulatory Aspects, Proceedings COST66 Symposium, Brussels*, 476-482.

Pestemer, W., Günther, P., Wischenewsky, M-B., Novopashenny, I., Wang, K. and Zhao, J. (1994) Development of expert systems to aid herbicide use with regard to their behaviour in soil. *Proceedings 1994 Brighton Crop Protection Conference - Weeds,* 1365-1372.

Trevisan, M., Capri, E., Evans, S.P. and Del Re, A.A.M. (1994). Validation and comparison of pesticide soil transport models for field dissipation of metamitron. In: *Environmental Behaviour of Pesticides and Regulatory Aspects, Proceedings COST66 Symposium, Brussels*, 501-508.

Walker, A (1994) Herbicide behaviour in soils in mediterranean climates. In: *Proceedings 5th EWRS Mediterranean Symposium*, pp. 211-221.

Walker, A and Hollis, J.M. (1994) Prediction of pesticide mobility in soils and their potential to contaminate surface and groundwater. In: *British Crop Protection Council Monograph*, No. 59, 211-224.

Pestemer, W and Günther, P. (1995) Use of a Pesticide Monitoring System (PEMOSYS) for the risk assessment of pesticide leaching potential: Concept for post-registration activities. In: *British Crop Protection Council Monograph,* No. 62, 351-356.

Trevisan, M., Capri, E., Del Re, A.A.M., Vischetti, C., Marini, M., Businelli, M., Donnarumma, L., Conte, E and Imbroglini, G. (1995). Evaluation of pesticide leaching models using three Italian data sets. *British Crop Protection Council Monograph,* No. 62, 269-274.

In Press

Del Re, A.A.M and Trevisan, M (1995) Selection criteria of xenobiotic leaching models in soil. *European Journal of Agronomy,* In Press.

Trevisan, M and Evans, S.P. (1995) Procedure per la valutazione di modelli matematici predittivi la contaminazione delle falde acquifere. *Ingegneria Ambientale*, In Press.

Walker, A., Welch, S.J., Melacini, A. and Moon Y-H (1996) Evaluation of three pesticide leaching models with experimental data for the herbicides atrazine, alachlor and metribuzin. *Weed Research*, 36, In Press.

2. THE EXPERIMENTAL PROGRAMME

2.1 THE DATA BASE

(U Baer, J-M Dabadie, INRA, Thiverval-Grignon, France;
A Melacini, A Walker, HRI, Wellesbourne, UK)

Results from 81 separate field experiments have been collected and presented in a standard format. They represent 24 from the United Kingdom, 23 from Italy, 17 from Germany, 10 from France and 7 from Sweden. Details of sites, soils, crops, pesticides, and experimental results are stored and specific weather data for each experiment are provided in computer files. A DOS directory and file structure was devised to permit uniformity of data presentation. An example of an experimental data file is given in Appendix I. A data base of information about the data sets was constructed and a software program (INDEX) written in the 4th Generation Language Clipper (Version 5.2) to run under DOS. This permits the data base to be browsed in various ways in order to examine and, if required, to select appropriate experimental data. All of the data sets are indexed according to location, experiment type, crop, pesticide, soil properties, pesticide properties, availability of laboratory and field data to characterise the pesticide-soil interactions, and availability of the weather data required for modelling purposes. The data sets have also been classified with the letter A, B or C to indicate their usefulness for model evaluation. The classification A indicates the most reliable, B indicates intermediate reliability and C indicates the least reliable. Data sets judged to be unsuitable for model evaluation have been excluded from the data base. INDEX uses a system of hierarchical Menus to enable the user to navigate through the available options. An outline of the system is shown in Figure 2.1. There are five options built into INDEX:

Update Records
Display Records
Delete Records
Update Contact Addresses
Backup All or Part of Data Base

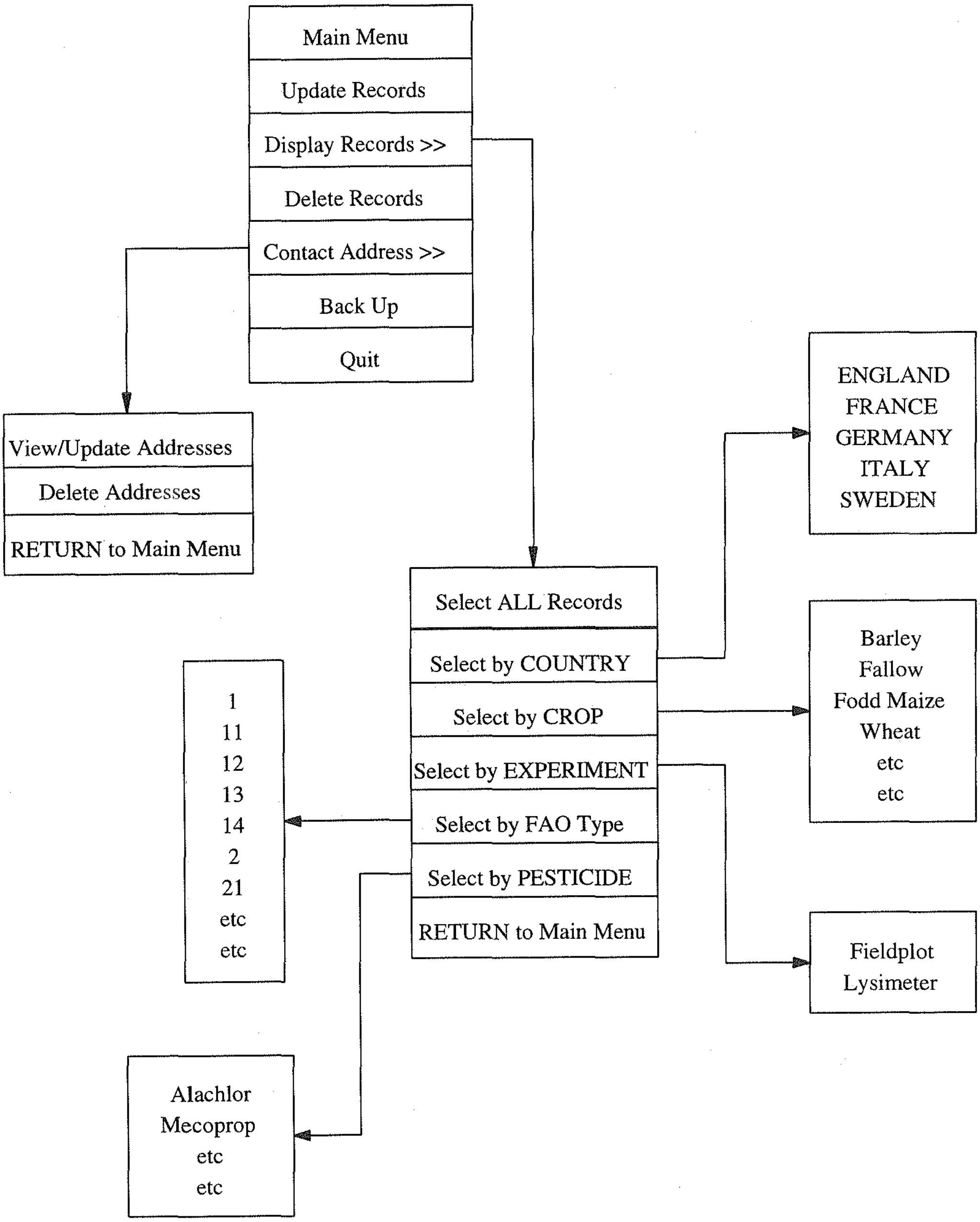


Fig. 2.1 Flow chart for the INDEX program

In the "Display Records" option, the system permits the Index to be examined in total or in part. When examined in part only, the sub-options permit display of records by country, experiment type, crop, pesticide, or soil type. For example, to display those records where the crop is barley, the user first selects "Display Records", then "Select by Crop" and finally select "Barley" from the list (Figure 2.1). When records are updated, any additional entry which introduces a new pesticide, crop, experiment type, soil type or country, automatically adds this new component to the relevant list used by the "Display Records" option.

Context sensitive Help is displayed at the base of the screen where appropriate.

2.2 MODELS AND MODELLING

2.2.1 Introduction

The main models used in the evaluation exercise were VARLEACH developed at HRI Wellesbourne and Rothamsted Experimental Station, UK; LEACHP developed at Cornell University, New York State, USA; and PRZM2 developed by the United States Environmental Protection Agency. Also available and used to a lesser extent were the simple screening model CMLS from the University of California, USA; the run-off model GLEAMS developed by the United States Department of Agriculture; and the MACRO model developed more recently at the University of Agricultural Sciences, Uppsala, Sweden. The latter describes preferential flow in soils prone to cracking. In order to simplify data input into the three main models (VARLEACH, LEACHP, PRZM2), a user-friendly interface was developed to permit "screen" input of data with suggested default values as appropriate and with an extensive help menu. A series of FORTRAN programs was also written to allow the weather data provided in the data base to be reformatted, as appropriate, for use in the various models. The models were then used in various ways in order to evaluate their performance.

2.2.2 Model Parameters

(P Günther & W Pestemer, BBA, Berlin, Germany)

Model input parameters and their availability

PRZM2

Weather data: Daily average air temperature (°C) and rainfall (cm). If available, pan-evaporation (cm), windspeed (cm/sec), solar radiation (Langleys) for simulating soil temperature. Evaporation can be estimated using monthly daylight hours and temperature, but the method used is simple and can be inaccurate. A pan factor is used to estimate daily evapotranspiration from pan evaporation. Windspeed and solar radiation are used only for simulation of soil temperature. Precipitation falling below 0°C is simulated as snow and a snowmelt factor (cm/°C) is used to calculate the speed of snow melting above freezing point.

Availability: These data, especially temperature and rainfall, are usually available, and evaporation can be measured or estimated (pan or potential). Windspeed and radiation are more difficult to obtain, but are often not needed for running the model.

Soil data: Field capacity, wilting point (can be calculated from clay and sand content), initial soil moisture content, bulk density (can be calculated from mineral density, if unknown), organic carbon content (%).
For simulation of biodegradation: growth, maintenance and death rates for different microbial populations
For erosion: factors for the universal soil loss equation, plot size (ha), average duration of storm rainfalls, runoff curve numbers for fallow, cropping, residue, universal soil loss cover management factors for fallow, cropping, residue (C values). Most of these values are available from the PRZM2-manual.
For irrigation: type of irrigation, water depth, application rate (cm/hr), flow rate in furrows (m^3/sec), width and slope of furrow walls and saturated conductivity of the soil, length, roughness and slope of the furrows.

Availability: Most of the basic soil data, which are used for a simple simulation without erosion, irrigation, biodegradation or volatilization, are available for most sites. Examples for some of the other input values, e.g. runoff curve numbers, soil reflectivity, surface albedo, are given in the PRZM2 manual. However, these values refer to US conditions, and their extrapolation to European scenarios is questionable. The data necessary for simulation of biodegradation have not been found for any of the experimental situations that we wished to investigate and these highly-specific values are almost impossible to obtain or estimate. This lack of usable degradation routines limits the usefulness of PRZM2.

Crop data: maximum interception storage, maximum rooting depth of the crop (cm),

maximum area coverage by the canopy (percent), surface condition of the crop after harvest (fallow, cropped, or residue), maximum canopy height at maturation date (cm), number of cropping periods, date of crop emergence, maturation, crop harvest.
Plant uptake of pesticides can be equal to transpiration and dissolved phase concentration, or a fraction of this, or not taken into account in the simulations.

Availability: These data should be available for most sites and can often be estimated. Some examples are also given in the manual.

Pesticide data: Application rate (kg/ha) and application depth (cm), initial pesticide levels before simulation (if applicable), surface condition for application (fallow, crop, or crop residue). For application to a crop canopy, maximum dry weight of the crop at full canopy (kg/m^2) is necessary. For the different application models (linear pesticide foliar application or foliar application using nonlinear exponential filtration) an additional filtration parameter is required. For pesticide decay on plant canopy, volatilization decay rate from canopy (1/d), decay rate on plant surface (1/d), foliar extraction coefficient for pesticide washoff per cm of rainfall are necessary.
Degradation: hydrolysis decay rates (1/d) in dissolved phase, in adsorbed phase and in vapour phase. For parent/daughter relationships, a transformation rate for one chemical to the other is necessary.
Adsorption: Distribution coefficient in soil (Kd). Can be estimated from Koc, water solubility (mg/l or μ mole/l), or mole fraction.
Volatilization: Henry Constant, diffusion coefficient in air, enthalpy of vaporization. Recommended default values are 0.43 m^2/d for the diffusion coefficient for all pesticides, and 20 kcal/mol for the enthalpy of vaporization.
Dispersion: hydrodynamic solute dispersion coefficient (cm^2/d). Default value 0.0.

Availability: For application to a crop canopy, example input values are given in the manual, but decay rates on plant surfaces are difficult to find. The field degradation rate can be estimated from field studies. If a lab DT$_{50}$ has to be used, a correction for the field conditions in question has to be performed. This must be done manually outside the model.

Kd / Koc should usually be available or easy to estimate within the program. Almost no experimental values are available for diffusion coefficient in air, enthalpy of vaporization, and hydrodynamic solute dispersion coefficient, so that default values are recommended.

LEACHP

Weather data: Daily rainfall (mm) and the time of day of rainfall, surface flux density: If the time of rainfall or the surface flux density is not known, both can be set to default values without affecting the results too much. Default for the surface flux density is 5 times the rainfall amount; smaller values limit infiltration.

Potential Evapotranspiration (weekly totals, mm), depth to water table (mm), mean weekly temperatures and mean weekly amplitude (°C). All these data can be obtained from daily minimum/maximum temperature, rainfall and pan evaporation data using the WEATHER.EXE delivered with LEACHP. Therefore the availability of the necessary weather data should be good, with the exception of evaporation.

Soil data: Lower boundary condition: fixed depth water table, free drainage, zero flux or lysimeter. For fixed water table, initial water table depth (mm) has to be specified. Clay, silt, bulk density, organic carbon content, initial soil moisture content (as potential or fraction), initial soil temperature, particle density for clay, silt, sand and organic matter (kg/dm^3), dispersivity. Water retention parameters (Campbell), saturated conductivity and matching matric potential.

Availability: Some of the soil data are often not readily available, e.g. soil retentivity parameters, saturated conductivity and dispersivity. Soil retentivity can be estimated using the implemented retentivity models, but for the soils examined in this project, they did not fit measured retentivity curves particularly well. Dispersivity is almost impossible to measure or to estimate, although default values can be used. The influence of the initial water content on the simulation result is usually negligible, because the soil water content adjusts within the first few rainfall events regardless of the starting value.

Crop data: Root fraction for each soil layer, date of crop germination, emergence, maturation, crop harvest, minimum root water potential, crop cover fraction, relative rooting depth. Plant uptake of pesticides can be equal to transpiration multiplied by dissolved phase concentration, or a fraction of this, or not simulated.

Availability: The dates should be available in most cases. The other data can often be estimated or could be taken from the tables in the PRZM2 manual.

Pesticide data: Application date, rate (mg/m^2) and application depth.
Adsorption distribution coefficient (Koc), or Freundlich Koc with coefficient, water solubility (mg/l). Vapour density (mg/l), molecular diffusion coefficient in water (mm^2/d) and in air (mm^2/d), air diffusion coefficient enhancement, adjustment values for Bresler's equation. Degradation and transformation rate constants (1/d), factor by which rate constants change per 10 °C increase (Q_{10}), temperature at which the input data rate constants apply, optimum temperature up to which Q_{10} relationship applies (from 0°C to this value) , maximum temperature to which rate constants decrease from optimum. High/low end of optimum water content range (air-filled porosity/matric potential), minimum matric potential for transformations (Kpa), relative transformation rate at saturation.

Availability: Most of the data for adsorption and degradation are usually available. The data for the influence of temperature and moisture on degradation can be set to estimated values, if unknown (e.g. $Q_{10} = 2$).

VARLEACH

Weather data: Daily minimum & maximum air temperature (°C), rainfall (mm), pan-evaporation, if available (mm). Otherwise evaporation is estimated by the program using equations derived by Linacre. Latitude (degrees) and altitude (metres) of the site are required for Linacre estimation of evaporation.

Availability: These data, especially temperature and rainfall, are usually available. Evaporation can be difficult, because in some countries, e.g. Germany and France, standardized evapotranspiration (above short, well-irrigated turfgrass) is usually available.

Soil data: Profile division (depth where adsorption, degradation, bulk density and field capacity change); moisture contents at field capacity (-5 kPa) and at -200 kPa; initial soil moisture content, bulk density.
Availability: The soil data required for VARLEACH are usually available or can be estimated.

Plant data: no plants simulated

Pesticide data: Application rate (kg/ha), adsorption distribution coefficient (Kd) with adsorption increment (change of Kd over time, if available), water solubility (mg/l).
For degradation, an approach developed by Walker & Barnes (1981) is used. Constants are used to characterize the influence of moisture and temperature: A (moisture intercept), B (moisture slope), E (activation energy, J/mol) and the standard temperature T (°C) at which A and B were measured.
If these are unknown, the same approach is used with default values for B and E, and back-calculation of A within the program. In this case, half-life (days), and moisture (%) and temperature (°C) at which half-life was measured are required.
Volatilization is not simulated
Availability: A, B, & E-values for degradation often are not available because they have to be calculated from costly laboratory degradation assays. Laboratory half-life values are a good substitute. The adsorption increment is not necessary (can be set to 0, if unknown), but improves simulation results, even if set to the proposed default value of 1/10 Kd.

Reference

Walker A and Barnes A (1981) Simulation of herbicide persistence in soil: A revised computer model. *Pesticide Science*, **12**, 123-132.

2.2.3 Marvel Interface

(P Siddons, J M Hollis, & C D Brown, SSLRC, Silsoe, UK)

Models generally require a large number of interdependent parameters (see Section 2.2.2). Selecting an incorrect value for a parameter or an incorrect combination of parameters often causes the model to crash. The error messages (if any) are frequently opaque and do not enable the user to identify the cause of the problem.

Information about the units used when setting parameter values is often buried in the user manual for the model, or not provided at all.

The format of parameter files is critical; an incorrectly placed comma or an extra blank line can also cause the model to crash, again with unhelpful error messages.

MARVEL (Model Attribute Relationship Validation and Entry Layout) is a user-friendly interface which was developed to address these problems and facilitate model usage by providing

- redundant parameter exclusion
- parameter range checking
- sample values for initial runs
- re-runs with minimal change
- graphical display of results
- context-sensitive help

Three models may currently be used with the MARVEL package: VARLEACH, LEACHP and PRZM2. The software is written in the 4th Generation Language Clipper (Version 5.2) and runs under DOS.

The entire package is driven by a series of databases. This permits modification of parameters without the need to re-compile the program. Similarly, the context-sensitive Help system may be modified simply by editing and re-indexing the relevant database.

A User Guide has been written to accompany the software.

MARVEL Menu System:

A summary of the basic structure of MARVEL is shown in Figure 2.2. A set of hierarchical Menus enable the user to navigate through the available options.
On start-up the Main Menu is presented, offering a choice of Model and a utility "Update databases".

Menu choice is via arrow keys to highlight an item and the Enter key to select the highlighted item. Selecting a Model from the Main Menu produces Sub-Menu1 which offers the same choice of actions for each model.

Sub Menu1 - Prepare Parameters

When this option is selected the user is first presented with a list of existing job files which provide starting parameter values. Each job has a unique number and the file may be updated or a new file created using the next available number. The same job number is used in naming output files. For example the LEACHP Job 6 parameter file is named LCHP006.PAR and the corresponding output files LCHP006.OUT and LCHP006.SUM. If there are no previous jobs prepared for this model, then default values will be set and the job number will be 001.

The next screen lists available files of Meteorological data and the user selects an appropriate file. Several screens then follow which enable the user to enter values for the parameters required by the selected Model.

The Title of each screen and the prompts for each parameter briefly describe what is required. Further information is provided by the context-sensitive Help system whenever the Function key F1 is pressed.

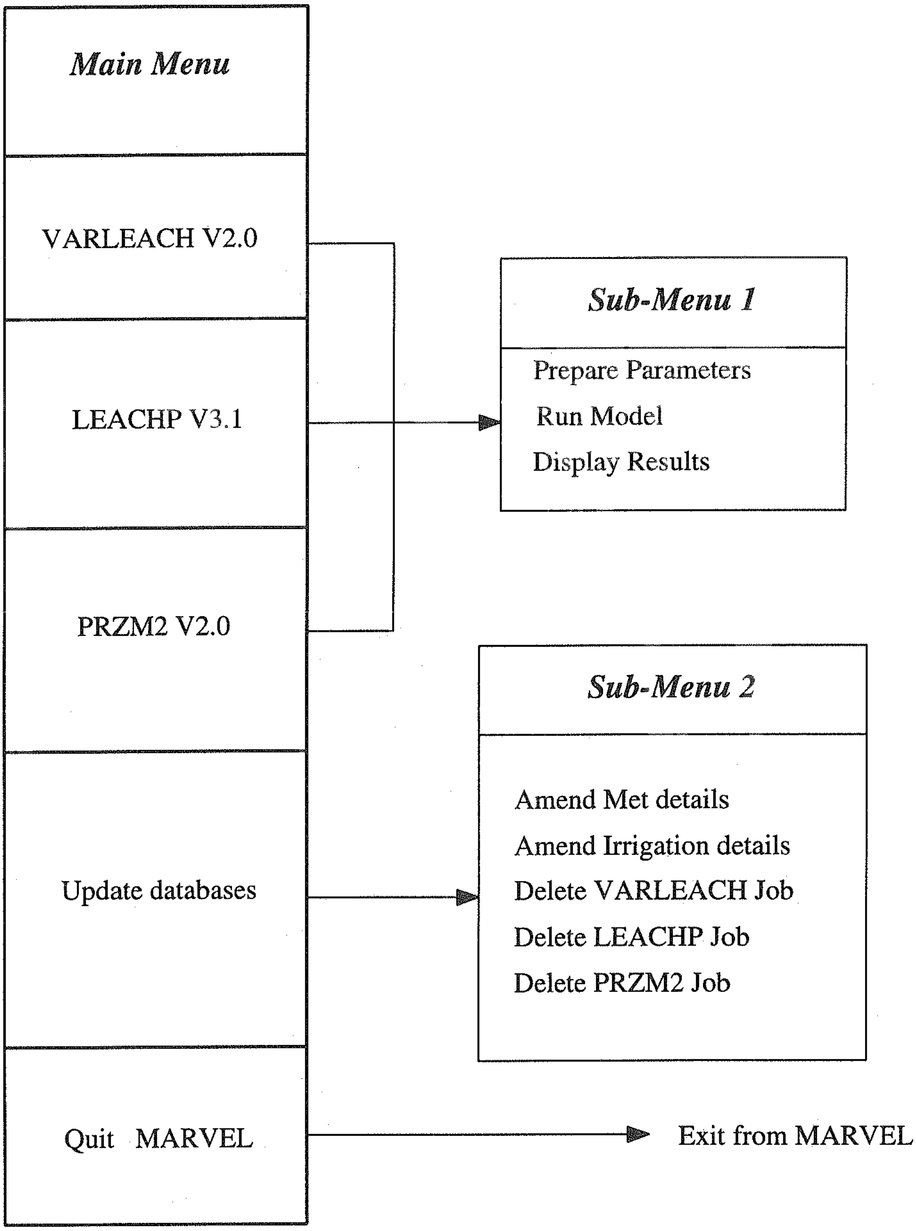


Fig. 2.2 Structure of the MARVEL program

Only required parameters need to be set and redundant parameters are not displayed. Parameter values are checked as they are entered and a warning message and the correct range is displayed for any parameters out of range.

Where repeated values are required, for example when entering data for 30 layers of a horizon, values are displayed in columns and any changes to a value within a column will result in the values further down that column being updated until a different value is encountered. This reduces the amount of typing required. Date values, however, are not modified in this manner.

The user may step backwards and forwards through the screens using the Function keys F2 and F3 until satisfied with all parameter settings. Parameters are then written to an ASCII file.

A line of information is displayed at the top of the screen at all times, showing the job number, the chosen Meteorological File, the current Screen number and a shorthand description of the Function keys available.

Sub Menu1 - Run Model

When this option is selected the user is presented with a list of prepared jobs (if any exist). When a Job has been chosen the package automatically prepares a batch file for running the relevant Model and then shells to the DOS prompt while displaying instructions to the user to run the batch file. Once the batch file commands are complete, MARVEL is restarted automatically at the Main Menu with the relevant Model highlighted.

Sub Menu1 - Display Results

When this option is selected the user is presented with a list of jobs which have been run (if any exist). Once a Job has been selected from this list, MARVEL reads the

output file and prepares graphical displays of the results.
The user is then offered a choice of these displays.

For example, PRZM-2 offers graphs of:

Cumulative water balance
Pesticide concentration in leachate (at base of profile)
Pesticide concentration in surface runoff
Pesticide concentration in eroded sediment
Daily losses of pesticide - in runoff and sorbed to sediment

If a local printer is connected, it is possible to obtain a screen dump of a graphical display.

Updating Databases

Selecting this option produces Sub-Menu2 which offers the choice of editing the databases which hold information about Meteorological and Irrigation data files, or deleting existing jobs.

When jobs are deleted, both the relevant parameter files and Model output files are removed. MARVEL may hold information on a maximum of 99 jobs and an unlimited number of Meteorological and Irrigation files.

2.2.4 Statistical Approaches

(A Melacini & A Walker, HRI, Wellesbourne, UK;
Petra Günther, BBA, Berlin, Germany)

Introduction

One component of a model evaluation exercise is to define criteria by which the agreement between observation and prediction can be assessed. As part of this research programme, various statistical indices were either taken from the literature or identified in discussions with statisticians. A description of the indices selected is given below. They were not all used in the subsequent modelling exercises, but they are listed here to give an indication of the types of index that are available.

The simplest way to assess model performance is to plot the predicted and observed values on a suitable diagram; this method represents an essential starting point for any model evaluation exercise. The visual method, although useful, needs to be backed up by some appropriate statistical evaluation so that relative model performance can be assessed in different situations. What is required is a system that enables us to estimate the fit (or the lack of fit) that is expressed graphically in the diagrams.

The criteria used in selection of the statistical indices were purposely chosen in relation to our specific context. Many alternatives were available, but only those with a simple and straightforward application were selected.

The Set of Indices

An index calculated from observed and predicted data can be used to express an overall fit of the model simulation or the fit of a particular aspect of the phenomenon we are modelling. In relation to the behaviour of pesticides in soil, the two main processes that determine the amount of a compound at a particular depth are:

1) movement of the pesticide through the soil
and 2) its degradation.

In order to analyze both the overall fit of the model prediction and the fit with respect to degradation and distribution, three different sets of indices were selected.

Notation

The evaluation process in the current research project deals mainly with the prediction of pesticide residues in soil from lysimeters, columns or field plots. The adopted notation is as follows:

Given a sample of *N* observations and their *N* corresponding predictions

O_i	Observed pesticide residue in the *i*-th layer *i*=1,..,N
P_i	Predicted pesticide residue in the *i*-th layer *i*=1,..,N
$\{O_i\}$	Sample of observations
$\{P_i\}$	Sample of predictions
iff	"*if and only if*" (logical equivalence: biconditional)
if..*then*...	logical implication.
$\forall i$	For all values (*i* = 1,...,N)

We will refer to the *graph* $\{(i,P_i)\}$ as the *Predicted Curve* (the plot of predicted values against depth), and similarly the *Observed Curve* refers to the plot of observed data against depth.

i) Evaluation of Overall Fit

Scaled Total Error

$$TE = \frac{\sum_{1}^{N}{}_{i} |P_i - O_i|}{\sum_{1}^{N}{}_{i} O_i} \qquad (1)$$

The numerator of the above ratio represents the ***total discrepancy*** between predicted and observed values. The reason for dividing by the total of the observed values is to scale the total error in relation to the size of the experiment. This enables us to use the same index (1) for comparison between different experiments (e.g. experiments that differ by the application rate).

Properties

Two properties follow by the definition of *TE*:

$TE \geq 0;$

$TE = 0$ *iff* $\forall i\ (P_i = O_i)$.

Scaled Root Mean Squared Error

$$SRMSE = \frac{1}{\bar{O}} \cdot \sqrt{\frac{\sum_{i,1}^{N} (P_i - O_i)^2}{N}} \qquad (2)$$

This quantity (2) is a measure of the *Spread* around the ideal case of $P_i = O_i\ \forall$i: With a perfect fit, the value $(P_i\text{-}O_i)$ would be zero. The left-hand factor in (2) is included in order to *scale* the *Root Mean Square Error* in a similar way to that used for Index (1).

Properties

Again, from the definition of (2) it follows that:

$SRMSE \geq 0$

$SRMSE = 0$ *iff* $\forall i\ (P_i = O_i)$.

Model Efficiency

This quantity is widely used in Model Evaluation exercises and is based on more sophisticated considerations than the previous indices.

$$ME = \frac{\sum_{1}^{N}{}_{i} (O_i - \overline{O})^2 - \sum_{1}^{N}{}_{i} (P_i - O_i)^2}{\sum_{1}^{N}{}_{i} (O_i - \overline{O})^2} \qquad (3)$$

In order to understand its behaviour, *ME* can be rearranged as follows (*ME* in this form is known as *Sutton-Rathcliffe's Coefficient*):

$$1 - \frac{\sum_{1}^{N}{}_{i} (P_i - O_i)^2}{\sum_{1}^{N}{}_{i} (O_i - \overline{O})^2} \qquad (3a)$$

Properties

The following observations can be made:

$ME \in]-\infty; +1]$ (*ME* has no lower bound and its upper bound is 1)

$ME = 1$ *iff* $P_i = O_i \; \forall i$

$ME \leq 0$ *iff* $(\sum_{1}^{N}{}_{i} (P_i - O_i)^2) / (\sum_{1}^{N}{}_{i} (O_i - \overline{O})^2) \leq 1$

The third property suggests that when *ME* becomes negative the fit is unacceptably poor. Under these conditions, the statistical sample variance of $(P_i\text{-}O_i)$ is less than (or equal to) the sample variance of $(O_i\text{-}\overline{O})$. Therefore accepting the model predictions is no better than simply using the mean of the observed data.

ii) Evaluation of the Prediction of Total Soil Residues

A number of indices can be used to evaluate the goodness of fit of the degradation simulation, disregarding the distribution of residues in the soil. The following indices look mainly at the prediction of the amount of pesticide in the profile relative to that observed.

Coefficient of Residual Mass

$$\mathrm{CRM} = \frac{\sum_{i=1}^{N} (P_i - O_i)}{\sum_{i=1}^{N} O_i} \qquad (4)$$

Properties

In this relationship:

CRM is neither upper nor lower bounded

CRM = 0 *iff* $\Sigma P_i = \Sigma O_i$ (perfect degradation fit)

CRM < 0 the model tends to ***under-predict*** soil residues

CRM > 0 the model tends to ***over-predict*** soil residues

Degradation Load

$$DL = \frac{\sum_{i=1}^{N} (P_i - O_i)}{\sum_{i=1}^{N} |P_i - O_i|} \qquad (5)$$

This is a measure of the weight of the error in the prediction of degradation relative to the total discrepancy. Index (5) is obtained by dividing index (4) with index (1).

Properties

DL ∈ [-1 ; +1]

DL = 0 *iff* no degradation error

DL > 0 *iff* residues are generally over-predicted

$DL < 0$ *iff* residues are generally under-predicted

$DL = \pm 1$ *iff* total over(+)/under(-) prediction ($\forall$i)

iii) Evaluation of the Prediction of the Distribution of Residues in Soil

The following indices reflect some aspects of the distribution of pesticide residues in the soil profile, irrespective of the absolute quantities of residues involved. These indices provide a closer investigation of the *similarities* (or *dissimilarities*) between the predicted and observed curves.

Coefficient of Determination

$$CD = \frac{\sum_{i \, 1}^{N} (O_i - \bar{O})^2}{\sum_{i \, 1}^{N} (P_i - \bar{O})^2} \tag{6}$$

This is the ratio between the spread of the observed values around their mean and the spread of the predicted values around the observed mean.

Properties

$CD \geq 0$

if $\{P_i\} = \{O_i\}$ *then* $CD = 1$

In general, the closer CD is to 1, the better the fit. When CD = 1, the statistical variance does not change if we use the predicted rather than the observed soil residues. There are therefore reasons to believe that there is good agreement between observed and predicted data.

WARNING

In this relationship, if the predictions differ from the observed only in their *position* along the profile but not in the amounts themselves, the index will still give a value of 1 i.e. although this is an example of ***bad distribution***, the index will assume a value of 1.

Coefficient of Shape

$$CS = \frac{\sum_{i=1}^{N} (O_i - \overline{O})^2}{\sum_{i=1}^{N} (P_i - \overline{P})^2} \tag{7}$$

This quantity (similar to *CD*) represents the ratio between the spread around their respective means of both the predicted and the observed values (i.e. ratio between the sample variance of the two sets). In graphical terms, this is reflected by a similarity in the shape of the predicted and observed curves.

Properties

Same properties as *CD*

$CS = 0$ ***iff*** $\exists$ constant $K > 0$ so that $\forall$i $(P_i = O_i \pm K)$

The second property means that when the predicted amounts differ from the observed amounts by a certain constant *K*, *CS* does not vary. This behaviour will be characterised visually as a predicted curve similar to the observed **but** shifted upwards or downwards by a constant factor. In this situation, the simulation of degradation is likely to be responsible for the overall lack of fit.

NOTE: As any index calculation is carried out on an equal number *N* (number of layers) of observations O_i against their predictions P_i, the condition of *good fit* for degradation ($\Sigma P_i \approx \Sigma O_i$) renders $CD \approx CS$. With these conditions, the information given by *CS* will be identical to that from *CD*.

Cumulative Value Test

The index *CVT* tests the simulation of the pesticide distribution down the profile.

$$CVT = \sum_{i=1}^{N} \left(\left| \frac{K(P_i)}{\sum_{i=1}^{N} P_i} - \frac{K(O_i)}{\sum_{i=1}^{N} O_i} \right| \right) \tag{8}$$

where, for a general distribution of values $\{V^i\}$

$$K(V_i) = \sum_{j=1}^{i} V_j \tag{8a}$$

This quantity is based on the comparison between two curves obtained after a *double transformation* of the predicted and the observed curve. The first transformation is carried out by the operator **K(·)**: when applied to the *i*-th value of the distribution, it returns the total cumulative amount of the residues (predicted or observed) from the top layer (*1*st layer) up to the *i*-th. The second transformation consists of dividing each cumulative **K(*Vi*)** by the total of the values along the profile **K(*V*$_N$)**. This step is necessary to *scale* each curve by the amounts involved in the simulation, therefore analyzing only the effect of the distribution (movement). In effect this gives the cumulative fraction with depth in the profile, and once more, represents the overall shape of the distribution curve.

Properties

$CVT \in [0\ ;\ N]$

$CVT = 0$ ***iff*** $\exists$ constant $\alpha > 0$ so that $P_i = \alpha \cdot O_i\ \forall i$

The first property states that this index is *always* positive. When the predicted curve differs ***proportionally*** from the observed along its entire length, we have the situation where the distribution is simulated satisfactorily and the total lack of fit *TE* (index 1) is due only to errors in prediction of the absolute levels of residues in the soil.

Mean Depth

$$MD = \frac{\sum_{i=1}^{N} r_i \cdot d_i}{\sum_{i=1}^{N} r_i} \tag{9}$$

Where r_i represents the soil residue in the i-th layer and d_i is the layer depth. This quantity gives a measure of the *penetration* of a compound into the soil. It is the only index that is *depth-dependent* and does not involve a direct comparison of observed and predicted data: its use consists of the comparison between the predicted and the observed Mean Depth (normally a ratio of the two). From a mechanical point of view, the Mean Depth represents the centre of mass of the pesticide distribution along the profile.

The Software STATIND.EXE

A computer program (running under standard MS-DOS) has been developed to calculate all of the above indices, and this is freely available to anyone who wishes to use it. It reads an input file whose records contain *Observed* and *Predicted* values (optionally the *depth* in order to determine *Mean Depth* index (9)), together with some specification for the particular run (lower tolerance limit, output file name, experiment title etc). The output file consists of a sequence of index summaries which refer to consecutive groups of records in the input file. During the input session, the user specifies the number of consecutive subdivisions of the data and their length(s). In addition, there is (as default) a *General Summary* containing the values of the two index sets *i)* and *ii)* (while the Partial Summaries contain all three sets).

<u>Note</u> The program has been developed to reflect the most usual situation of experimental data sets consisting of measured soil residues along a profile (subdivided into several layers) at different sampling times.

When the input file columns represent the observed and the predicted values in different layers, and each record group refers to a sampling time, the interpretation of the information reported in the output file is straightforward. The Partial Summaries show the values calculated for the indices at each sampling time, whereas the General Summary gives information for the whole experiment. Examples of use of the statistical indices are given in Section 2.2.5 below.

2.2.5 Comparison with data sets

*(P Günther, BBA Berlin, Germany; A Walker, HRI Wellesbourne, UK;
M Trevisan, UCSC Piacenza, Italy; C D Brown, SSLRC Silsoe, UK; R Calvet,
INRA, Thiverval-Grignon, France)*

2.2.5.1 VARLEACH, LEACHP & PRZM2

Selected data sets taken from the main data base were used to evaluate the predictive ability of various mathematical models of pesticide fate. The main models used were VARLEACH, LEACHP and PRZM2. Details of the background to the three models are given by Walker *et al* (1996). Appropriate soil characteristics such as the relationships between soil water tension and soil water content, mechanical composition, organic matter content and bulk density, combined with constants describing factors such as adsorption, degradation rate and potential volatilisation of the pesticides provide the input parameters for the three models. All of these parameters were taken from the data base (eg. Appendix 1). Also required were appropriate weather data, again provided in the data base. Examples of parameterisation of the models are given by Walker *et al* (1996) and Trevisan *et al* (1995). These papers also provide detailed discussion on the accuracy of the predictions and use the statistical program STATIND as an aid to evaluation of results. Examples of comparisons between observed and predicted results are shown in Figures 2.3 to 2.9. These results have been selected to illustrate data relevant to experiments in the UK (2.3, 2.4, 2.5), Germany (2.6, 2.7), France (2.8) and Italy (2.9).

Using UK data: Evaluations concentrated on 6 data sets involving different herbicides, soils and experimental procedures. All three models generally gave acceptable predictions of the distribution of residues in the soil, particularly for the less mobile compounds. There was however considerable variation between models and between specific data sets in terms of the closeness of agreement, and the relative predictive ability. Examples were obtained where each of the three main models out-performed the other two, but there was no clear indication of the reason for this. Results from the UK modelling work suggest that the models can be used to estimate overall soil residues and their distribution in soils, but that none of them can be relied upon to do this in all

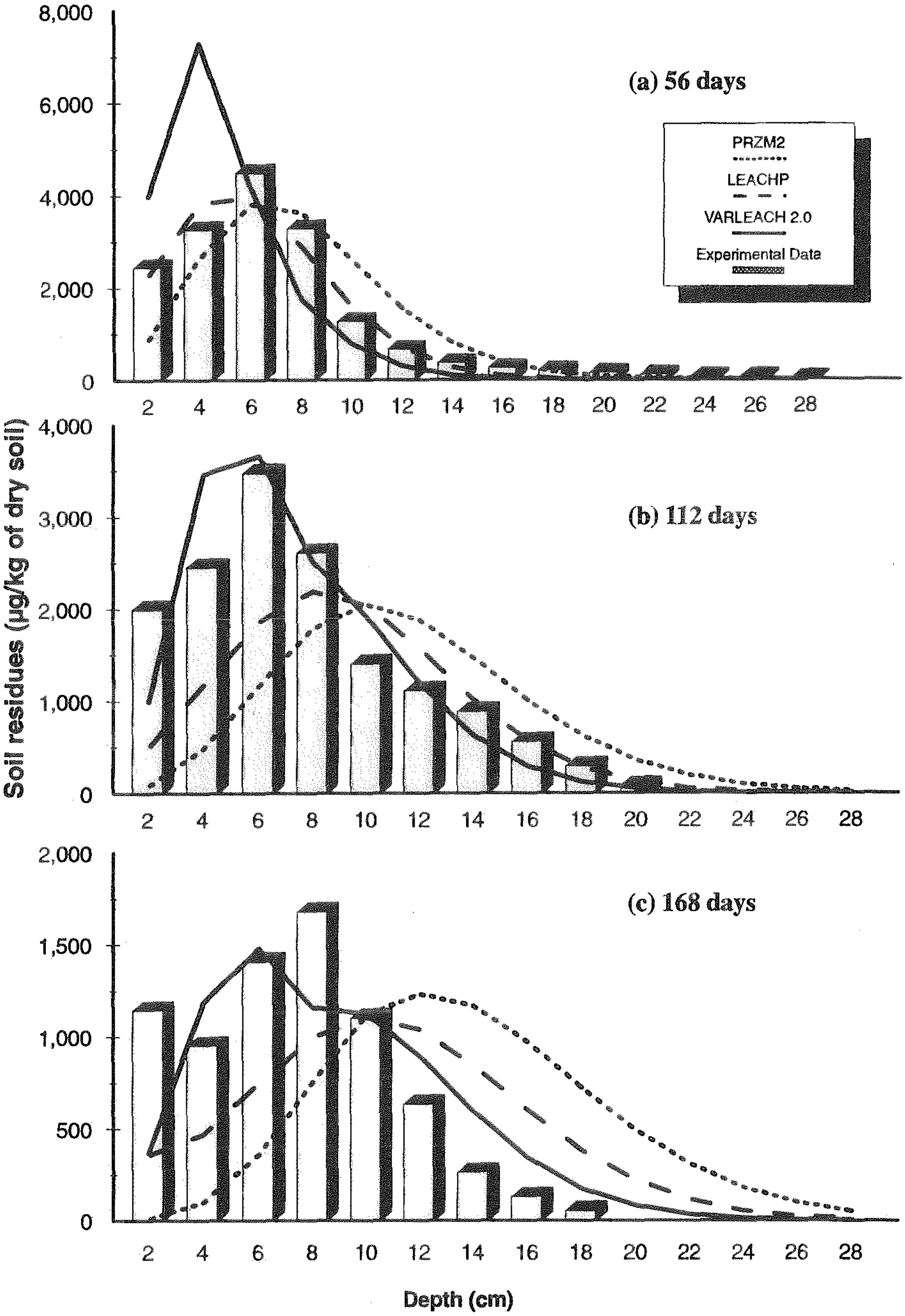


Figure 2.3 Measured and predicted residues of alachlor in a UK lysimeter study

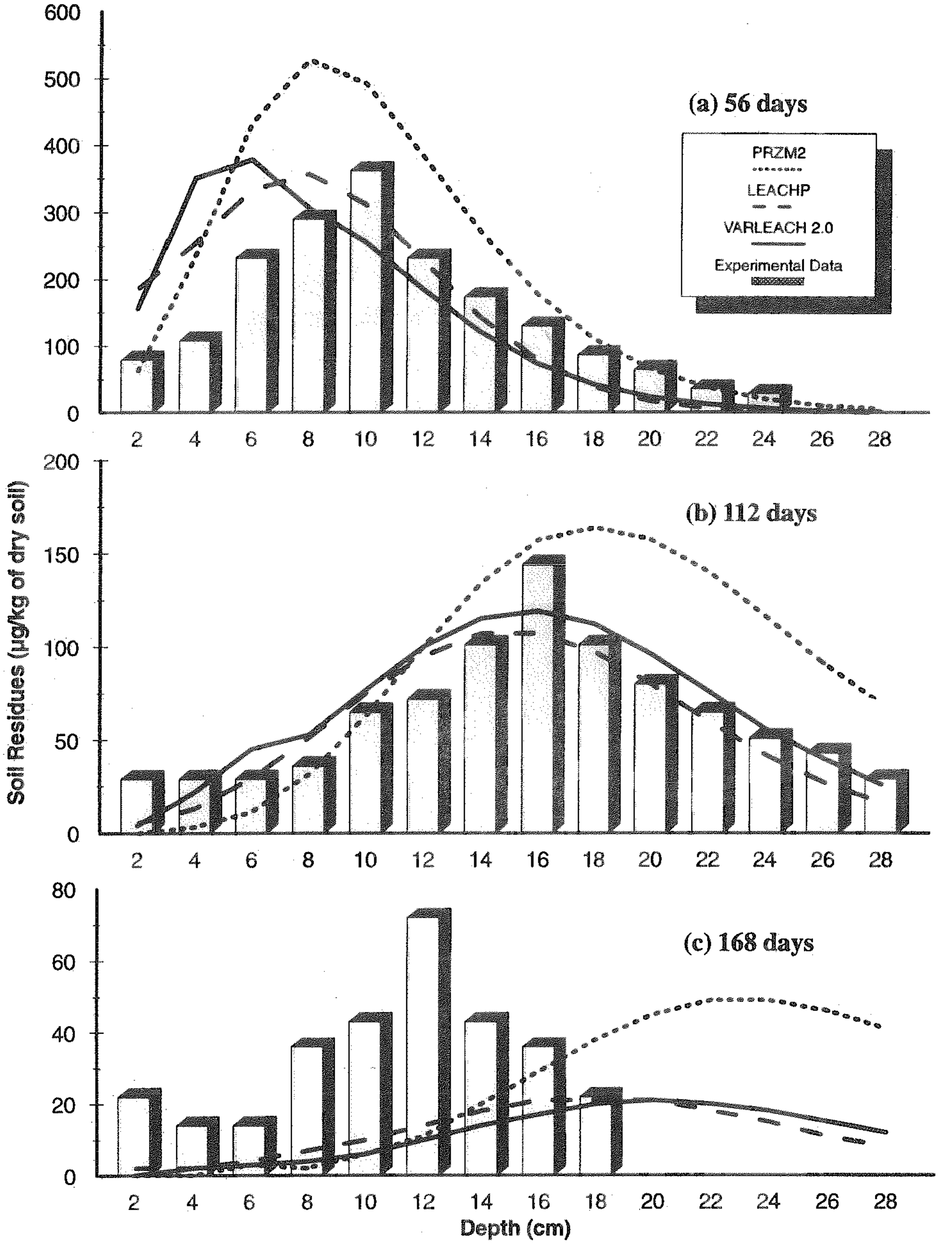


Figure 2.4 Measured and predicted residues of metribuzin in a UK lysimeter expt

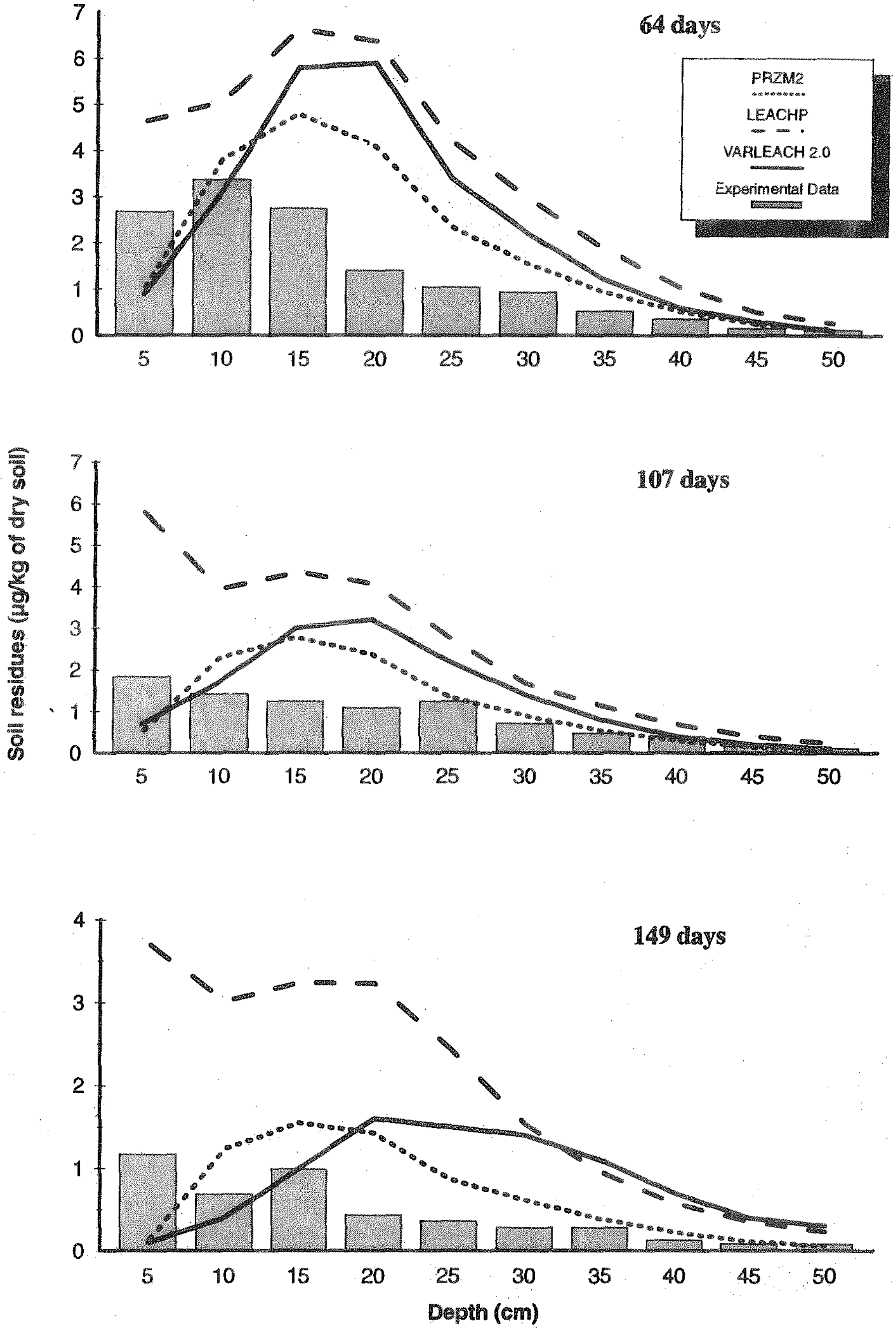


Figure 2.5 Prediction of metsulfuron-methyl mobility in a UK field expt

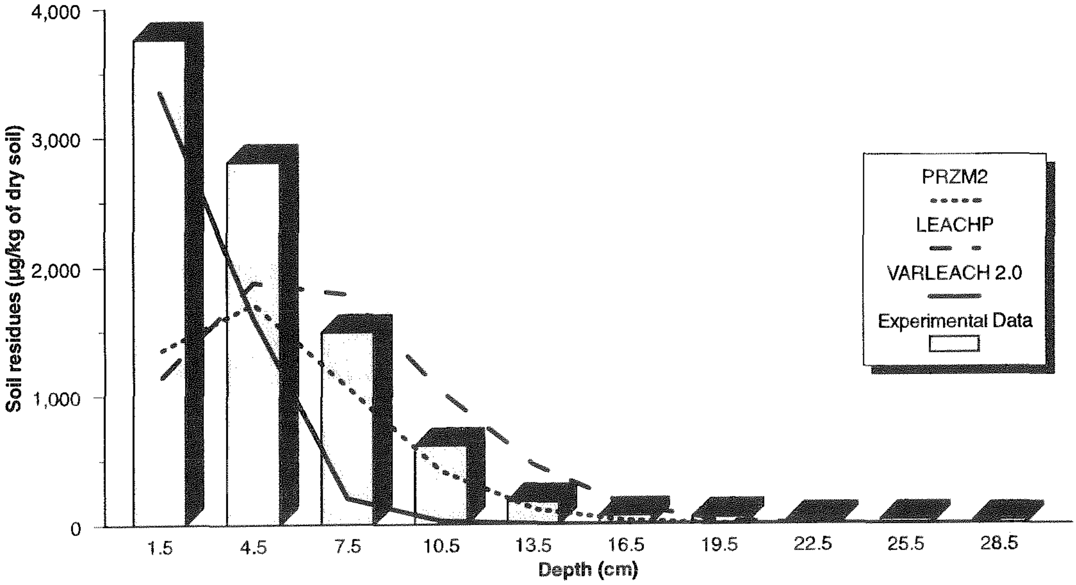


Figure 2.6 Prediction of terbuthylazine residues in soil 143 days after application in a German field study

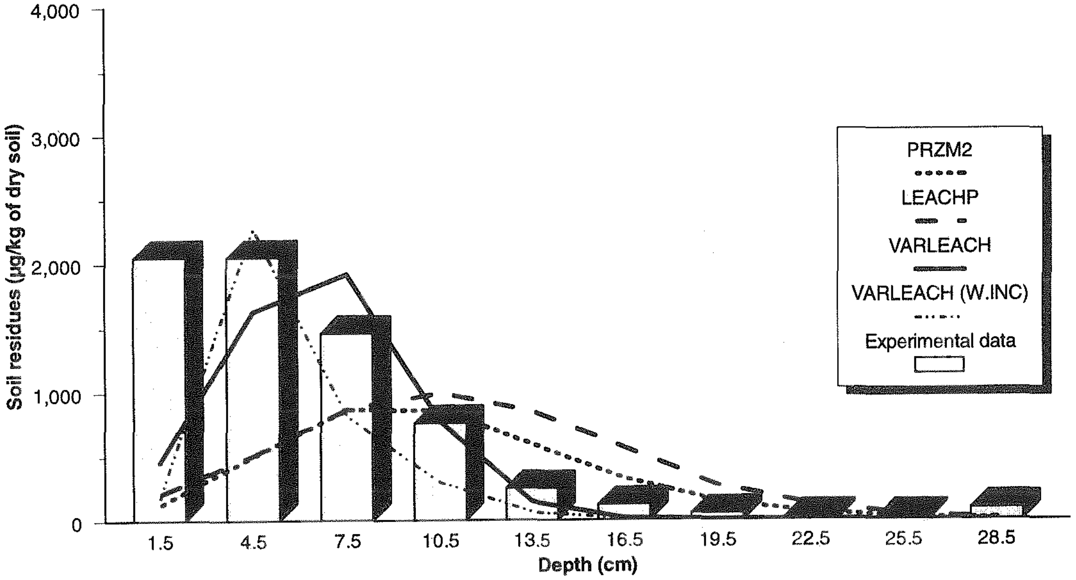
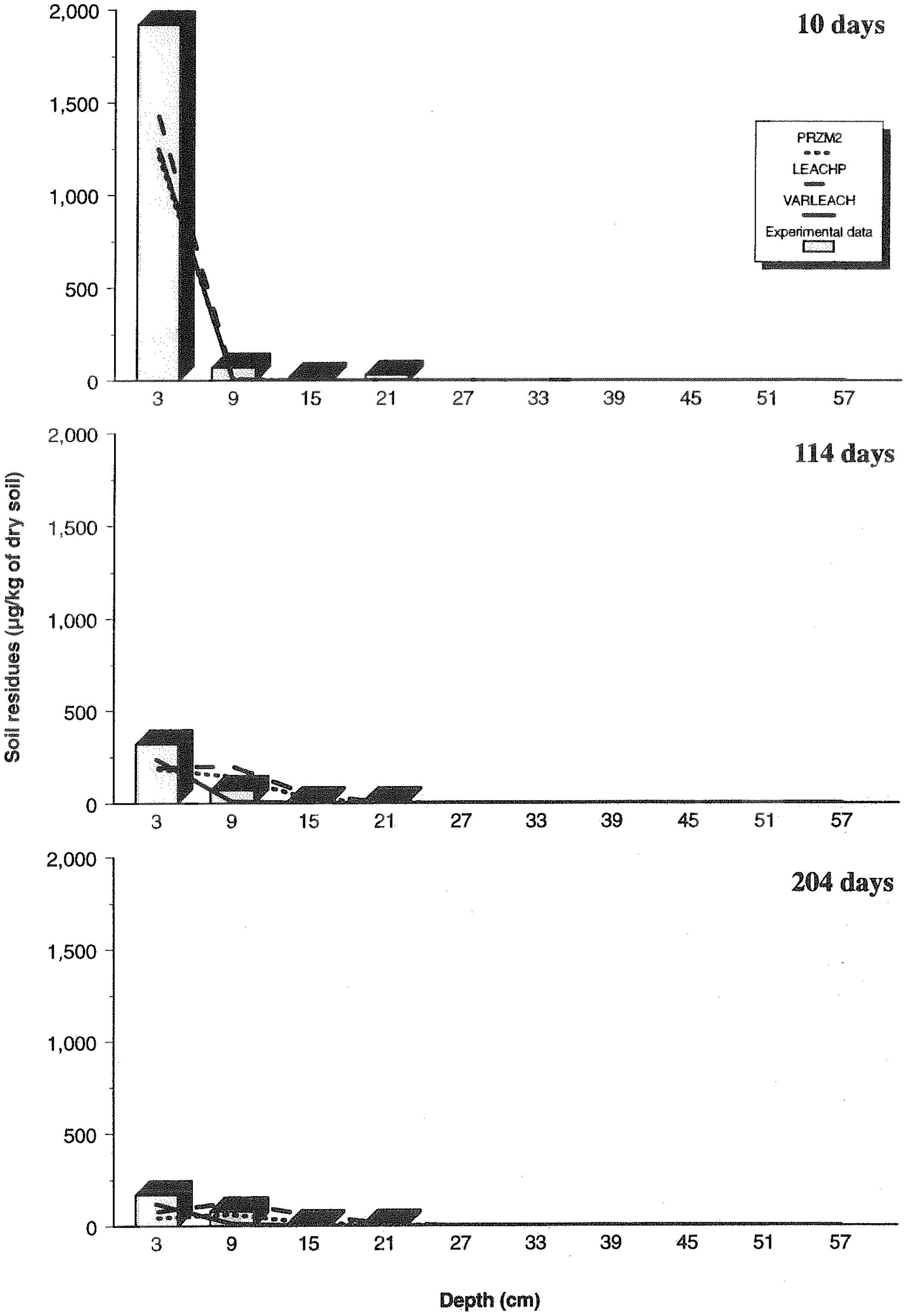


Figure 2.7 Effect of sorption increment on VARLEACH predictions, and comparisons between models for prediction of terbuthylazine residues 220 days after application in a German field study


2,000
1,500
1,000
500
0
10 days
PRZM2
LEACHP
VARLEACH
Experimental data
3
9
15
21
27
33
39
45
51
57
114 days
204 days
Soil residues (µg/kg of dry soil)
Depth (cm)

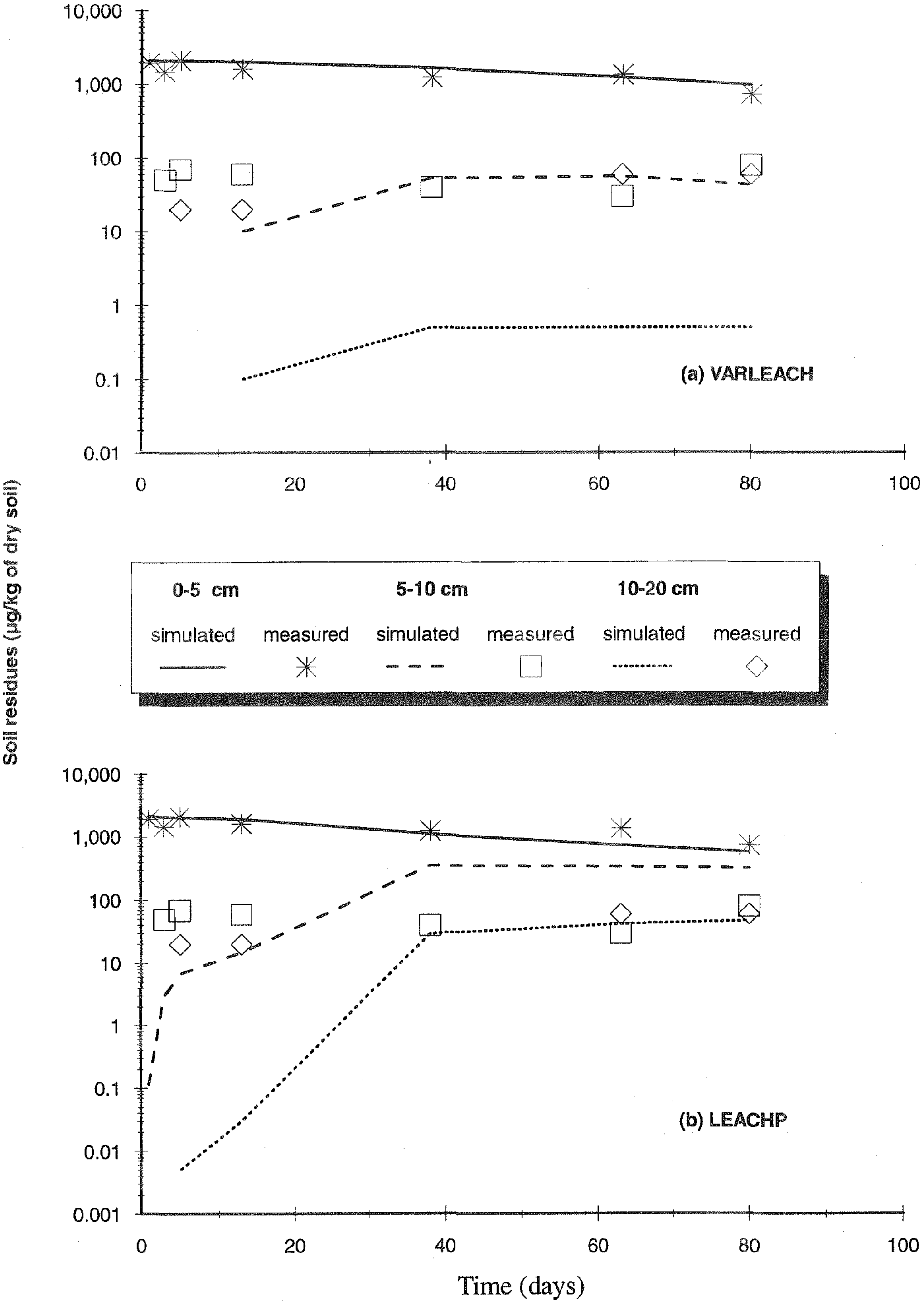


Figure 2.9 Measured and predicted chloridazon residues in a field experiment in Italy

circumstances. Further examination of model output data, when tested against results from studies in small lysimeters, indicated that all three models could give accurate predictions of the volumes of drainage water but that none could predict concentrations of pesticide in the leachate water.

Using data from Germany and France : For 10 German and 6 French field trials (25 and 38 sampling dates respectively), simulations with the models LEACHP, PRZM2 and VARLEACH were carried out and the results were compared with the measured residues. The simulation of overall fit was assessed using model efficiency, calculated with the STATIND software (Section 2.2.4) In addition, degradation and leaching depth were assessed separately. The total residues simulated in the profile were compared with those measured; for 3 French data sets there were also repeated measurements of residues in the top 10 cm over time. The fit of the distribution was assessed using mean leaching depth ratio, calculated with the STATIND software. Pesticide leaving the profile was negligible in all trials. The results of these assessments for the three models are shown in Table 2.1 and Figure 2.10.

The simulations were considered as sufficiently correct if model efficiency (ME) was above 0.6. This was the case in only 38% of all simulations. Differences between the models were not great (Figure 2.10); LEACHP gave satisfactory results in about 38% of the comparisons, PRZM2 was slightly better (40%) and VARLEACH slightly worse (35%). There was however a difference between German and French data sets regarding model performance. For the French data, PRZM2 gave by far the best fit with 49% acceptable simulations, followed by LEACHP with 39% and VARLEACH with 37%. For the German data, LEACHP (36%) gave the best fit, whereas VARLEACH (32%) and PRZM2 (28%) were less accurate.

The simulation of degradation was considered correct if total residues in the profile were estimated within 20% of the measured values. Of all simulations with both the French and the German data sets, only 23% were accurate, with LEACHP showing a slightly better performance than the others. There was no difference in the accuracy of simulations derived using data from either country, even though all German data sets had laboratory DT_{50} values whereas most of the French data had field half lives.

Table 2.1 Overall assessments for all models and simulations (as a percentage of 189 individual simulations)

	Model Efficiency[1]			Degradation[2]			Depth Ratio[3]		
	All	Ger	Fr	All	Ger	Fr	All	Ger	Fr
Overest	**	**	**	35	35	35	21	21	21
OK	38	32	41	23	24	22	46	43	48
Underest	**	**	**	42	41	43	33	36	31

[1] Model Efficiency: > 0.60 = OK

[2] Degradation: 20% variation around measured total residues = OK

[3] Assessment of Leaching: 0.8 < Depth Ratio < 1.2 = OK

** Model Efficiency results cannot be evaluated with respect to over or under-estimation of results

Residues in the top soil layer (0-10 or 0-20cm) were compared with modelling results for three French data sets. In two of the trials the fit was very good; however, in the third the predicted initial concentration was much too high, although the fit improved with time.

The simulation of pesticide distribution in soil was slightly better than for total residues, but still not satisfactory. Distribution was considered correct if the ratio of mean leaching depths for measured and simulated data was between 0.8 and 1.2 (20% difference). It must be recognised, however, that this approach will not always correspond with visual assessments, especially where most of the residues remain in the top soil layer. An example is given in Figure 2.6, where PRZM2 was the only model which predicted an acceptable mean leaching depth, although VARLEACH gave an improved visual estimation. Overall, satisfactory estimates of distribution were obtained in 46% of all simulations. PRZM2 showed the best performance, with 51% of distributions correctly simulated, followed by

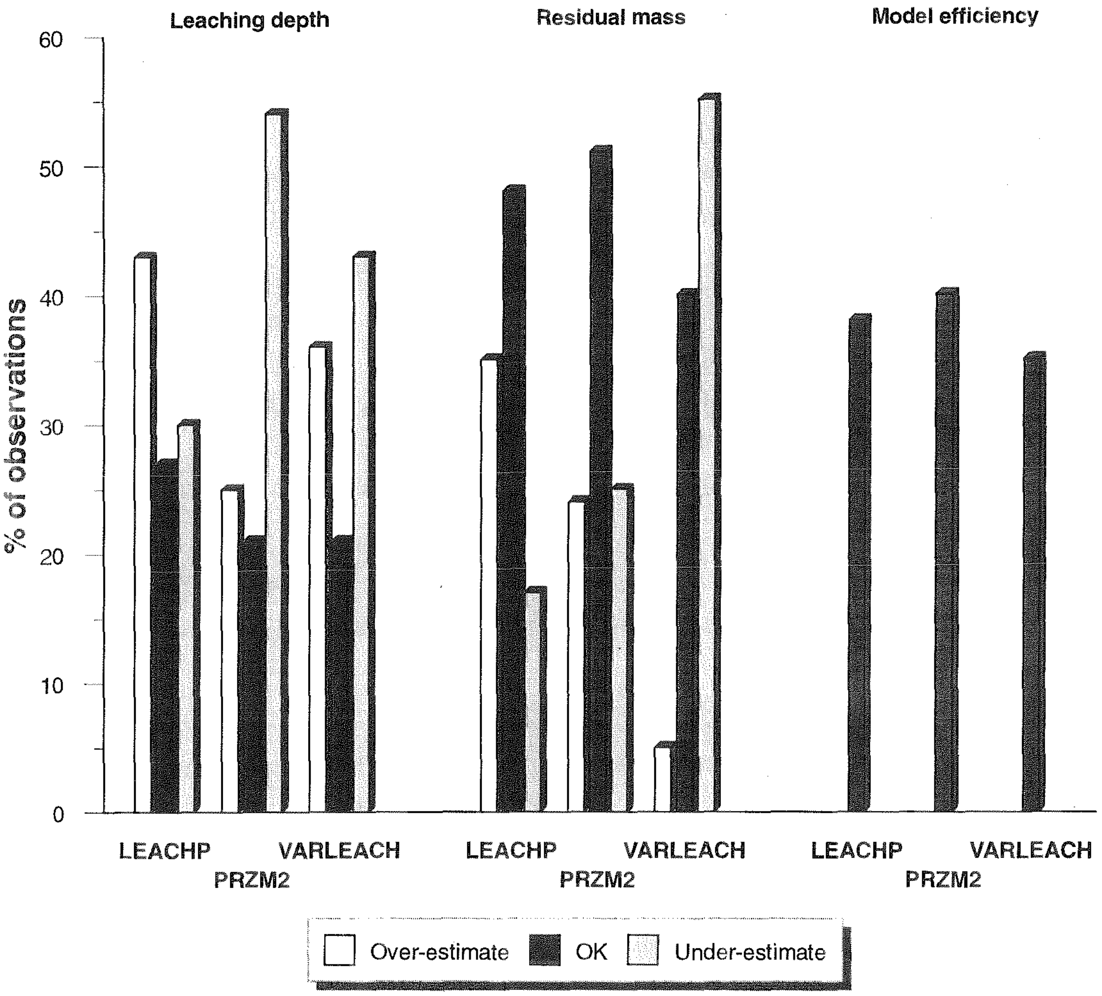


Figure 2.10 Accuracy of model predictions in 63 comparisons involving data from French and German field experiments

LEACHP (48%) and VARLEACH (40%). For the German data, PRZM2 gave the best fit, with 52% of distributions correctly simulated. For the French data, PRZM2 and LEACHP both provided accurate predictions of distribution in 50% of all simulations; LEACHP tended to overestimate leaching depth (35%), whereas VARLEACH underestimated leaching depth (55%). There was no specific trend with PRZM2.

With regard to leaching simulations, there was again no great difference between German and French data, although there was a slightly greater tendency to underestimate leaching using the German data (36% compared to 31%). This difference occurred mainly in simulations produced by LEACHP and VARLEACH.

Looking at single datasets, VARLEACH predicted less leaching than LEACHP or PRZM2. This was partly due to the increase of adsorption over time, as seen in Figure 2.7. Therefore the difference between the models was pronounced when compounds were strongly adsorbed. Accuracy of the weather data may also influence results. For all of these runs, only estimated potential evaporation was available, whereas VARLEACH was written to use measured pan evaporation. No correction factor can be applied by the model user.

When simulating experiments with repeated soil samplings, a change in the goodness of fit during the trial was observed. Usually the best fit was obtained during the first one or two months of the simulation, and during this time the results of the different models were quite similar (Figure 2.8). This is because the effects of any differences in the simulation procedure will be enhanced during the course of the simulation. Particularly in long trials with many sampling dates, results alternated several times between good fit and over- or underestimation, with no predictable pattern. This may result partly from variation in field residues, which often do not show a continuous decrease over time compared to model simulations. However, although a range of variation occurred in some of the French data, the variation within model results was still greater.

Using Italian data: Four data sets were examined in detail, and examples of results are shown in Figure 2.9. The most extensive evaluations were made using data for metamitron behaviour in soils at three contrasting sites (full details in Trevisan *et al.,* 1995).

Predictions after a standard time period were compared with observations using all of the indices described in Section 2.2.4. The results are summarised in Table 2.2, in which the values for the various indices are presented. The best value for each index/site combination is indicated in bold type. The results illustrate the extreme variability between data sets in the apparent "best fit model". For these data, VARLEACH generally gave the best predictions of the data from Perugia, LEACHP gave the best predictions for the Piacenza data set, and PRZM2 gave the best predictions for the data from Rome.

Table 2.2 Statistical Indices for the three models VARLEACH (VA), LEACHP (LE) and PRZM2 (PZ) when tested against data from Perugia, Piacenza and Rome. Values in bold indicate the best value for the index/site combination.

Index	Perugia			Piacenza			Rome		
	VA	LE	PZ	VA	LE	PZ	VA	LE	PZ
TSE*	**0.456**	0.561	1.397	**0.503**	0.505	0.812	2.389	1.478	**0.808**
SRMSE	**0.645**	0.685	1.853	0.805	**0.759**	1.171	4.637	2.577	**1.151**
ME	**0.787**	0.760	0.757	0.705	**0.738**	0.376	-7.755	- 1.708	**0.460**
CRM	**0.166**	0.390	1.342	0.446	**0.313**	0.473	2.242	1.479	**0.807**
DL	**0.366**	0.696	0.960	0.886	0.619	**0.582**	**0.939**	1.000	1.000
CD	0.476	**0.568**	0.252	**2.500**	2.756	4.058	0.067	0.156	**0.544**
CVT	0.281	**0.187**	0.256	**0.111**	0.187	0.383	0.075	**0.040**	0.275
MD	0.781	**0.943**	1.092	0.910	**1.053**	1.105	0.930	**1.037**	1.256

*Total scaled error, scaled root mean squared error, model efficiency, coefficient of residual mass, degradation load, coefficient of determination, cumulative value test and mean depth. Definitions of indices are given in Section 2.2.4

2.2.5.2 Other models

A GLEAMS

The management model, GLEAMS, which was developed in the US to simulate runoff and erosive losses of chemicals was evaluated against data collected at Temple Balsall, UK. In the experiment, alachlor was applied to a sandy loam and a clay loam soil over a 3-year period and runoff was collected using 1m^2 traps and analysed for pesticide content.

Results from the field experiment and their simulation with GLEAMS are summarised in Table 2.3. There was considerable variability in the predictive ability of the model, with one of the three parameters well simulated, whereas the remaining two were under- or over-predicted by up to two orders of magnitude. No pattern occurred in the difference between observed and simulated results, either between years or between soil types. GLEAMS has been validated and has been extensively used in the US. The model utilises the runoff curve approach and universal soil loss equation to generate runoff and erosive losses, respectively. However, extrapolation of this approach to UK conditions and selection of appropriate input parameters is difficult (Table 2.3). It can be concluded that considerable validation and/or modification will be necessary before this model can be applied to European situations with confidence.

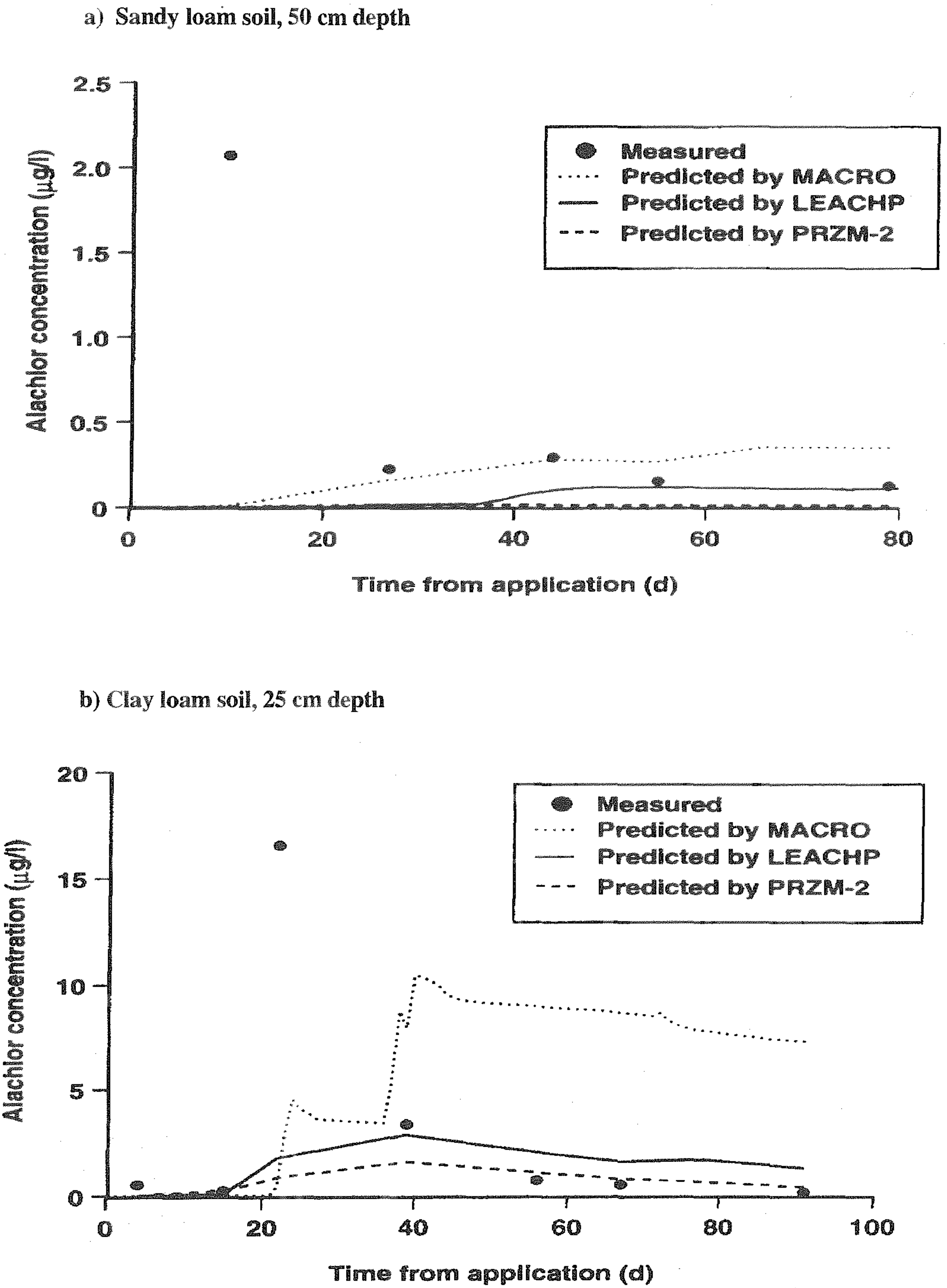
Table 2.3 Measured and predicted runoff and associated losses of alachlor

Parameter over experimental	Season 1		Season 2		Season 3	
period	Measured	Predicted	Measured	Predicted	Measured	Predicted
Sandy loam soil						
Total runoff (mm)	2.5	2.9	3.1	0.1	3.7	1.7
Maximum conc. of alachlor (μg/l)	1960	804	114	132	12	57
Total loading of alachlor (μg/m^2)	1303	567	196	6	16	12
Clay loam soil						
Total runoff (mm)	2.8	10.0	1.5	28.3	10.6	9.6
Maximum conc. of alachlor (μg/l)	161	386	123	226	12	145
Total loading of alachlor (μg/m^2)	153	1192	310	2346	10	269

B MACRO

MACRO is a mechanistic model which divides soil into macropore and micropore regions in order to describe preferential flow of water and associated solutes. The model was used by its author, Prof. Nick Jarvis, to simulate the observed leaching of alachlor in experiments at Temple Balsall on a sandy loam and clay loam soil (Jarvis *et al.*, 1995). The results of the simulation are compared with observed concentrations of alachlor in soil water and with simulations of the same experiment, using LEACHP and PRZM2, in Figure 2.11. For the sandy loam soil (Figure 2.11a), there was relatively little difference between simulations made with MACRO and with LEACHP. This is because preferential flow is initiated in MACRO when the conductive capacity of the micropore region is exceeded - a condition which will occur only rarely in coarse-textured soils. Thus both MACRO and LEACHP solve Richards' equation for a single pore domain, with both found to adequately predict leaching to 50-cm depth, with the exception of the isolated high concentration observed 10 d after application. PRZM2 did not perform as well as the other two models with this soil type.

Differences between MACRO and models which do not simulate preferential flow are likely to be greatest for fine-textured soils - where this flow process can be very significant. This is confirmed in Figure 2.11b, which compares observed and predicted leaching in the clay loam soil. MACRO predicted significantly more leaching to 25-cm depth than either LEACHP or PRZM2 and was the only one of the three models able to simulate the maximum concentration of alachlor observed. However, MACRO was found to over-predict observed leaching in the latter part of the experiment compared to the other two models. Preferential flow has been shown to have a significant impact upon leaching of pesticides in the field, and modelling this process will represent an important advance in simulating observed chemical leaching behaviour. The results of the limited evaluation undertaken here suggest that MACRO provides enhanced ability to simulate leaching in fine-textured soils, whilst retaining a mechanistic solution of Richards' equation similar to that of LEACHP under conditions in which preferential flow is not thought to be a significant process.



a) Sandy loam soil, 50 cm depth

b) Clay loam soil, 25 cm depth

Figure 2.11 Comparison between observed concentrations of alachlor in soil water and those predicted by MACRO, LEACHP, and PRZM2 for two field studies

References:

Trevisan, M., Capri, E., Del Re, A.A.M., Vischetti, C., Marini, M., Businelli, M., Donnarumma, L., Conte, E and Imbroglini, G. (1995). Evaluation of pesticide leaching models using three Italian data sets. *British Crop Protection Council Monograph*, No. 62, 269-274.

Jarvis, N.J., Larsson, M., Fogg, P.F. & Carter, A.D. (1995). Validation of the dual-porosity model "MACRO" for assessing pesticide fate and mobility in soils. In *Pesticide movement to water* (eds. A.. Walker *et al.*), pp. 161-170. BCPC Monograph No. 62, British Crop Protection Council, Farnham, Surrey.

Walker, A., Welch, S.J., Melacini, A. and Moon Y-H (1996) Evaluation of three pesticide leaching models with experimental data for the herbicides atrazine, alachlor and metribuzin. *Weed Research*, 36, In Press.

2.2.6 Sensitivity Analyses

(P Günther & W Pestemer, BBA Berlin, Germany)

Sensitivity analysis was carried out with the three main models LEACHP, PRZM2 and VARLEACH. Not all of them use the same input parameters so that results are not always comparable. The basic input data used in the analyses are listed below:

All simulations were based on a Standard Scenario (Ahlum 1 - data set GE01 in the data base) with the following input values:

Output: 220 days after application (last sampling date).
Profile depth: 30cm
Segment thickness 3cm
Lower boundary condition: free drainage
no plants during simulation
no residues before simulation

							PRZM2 only	
Layer Point	**Clay**	**Silt**	**Org C**	**Start Theta**	**Roots**	**Start temp**	**Field Cap.**	**Wilt.**
No	**%**	**%**	**%**		**%**	**°C**	**W/V**	**W/V**
1	12.9	78.3	1.4	0.257	.25	10.	34.9	9.2
2	12.9	78.3	1.4	0.257	.25	10.	34.9	9.2
3	12.9	78.3	1.4	0.257	.25	10.	33.7	10.8
4	11.2	81.3	1.4	0.257	.25	10.	33.7	10.8
5	11.2	81.3	1.4	0.257	.25	10.	34.1	7.0
6	11.2	81.3	1.4	0.257	.25	10.	36.5	9.3
7	12.8	79.8	1.2	0.257	.25	10.	38.1	9.2
8	12.8	79.8	1.2	0.257	.25	10.	38.1	9.2
9	12.8	79.8	1.2	0.257	.25	10.	38.1	9.2
10	12.8	79.8	1.2	0.257	.25	10.	38.1	9.2

Layer	**Soil retentivity par.**		**Bulk density**	**Match K(h) curve at:**			**Dispersivity**
	AEV	**BCAM**		**K**	**Matric pot'l**		
No	**kPa**		**kg/dm3**	**mm/d**	**kPa**	**P**	
1	-0.86	5.97	1.38	19	-00.	1.0	10.0
2	-0.86	5.97	1.38	19	-00.	1.0	10.0
3	-15.90	4.02	1.63	19	-00.	1.0	10.0
4	-15.90	4.02	1.63	19	-00.	1.0	10.0
5	-20.30	3.20	1.63	19	-00.	1.0	10.0
6	-17.80	3.62	1.63	19	-00.	1.0	10.0
7	-16.60	3.41	1.63	19	-00.	1.0	10.0
8	-16.60	3.41	1.63	19	-00.	1.0	10.0
9	-16.60	3.41	1.63	19	-00.	1.0	10.0
10	-16.60	3.41	1.63	19	-00.	1.0	10.0

For VARLEACH:
Bulk density = 1.55 (average)
Field capacity w/w (-3 kPa): 26.0% , below 6 cm: 23.4%
Water content -200 kPa (est.): 15.6%

Chemical parameters (Terbuthylazin)
Solubility 8.5 mg/l
Application: 1.9.1988, 5 kg/ha
Vapour Density 1.41E-05 mg/l
Freundlich isotherm: K_{oc} 121.4 l/kg, Exponent 0.893
K_{oc} 121.4 l/kg
K_d: 1.7 l/kg, adsorption increment 0.17
Rate constants apply to bulk soil
Rate constant (transformation): 0.0102/d / DT_{50} = 68 d at 26% moisture/20 °C
Field dissipation rate constant (for PRZM2) 0.005 (DT_{50} = 133 d)
***For LEACHP*:**
Temperature subroutine and adjustments included
Q_{10} = 2
Optimum temperature: 35 °C
Maximum temperature: 50 °C
High end of optimum water content range: air-filled porosity 0.08
Lower end of optimum water content: -33 kPa
Minimum matric potential for transformations -1500 kPa
Relative transformation rate at saturation: 1

Results of the Sensitivity Analysis

Chemical parameters

The chemical properties of the compounds used for simulation do not differ much between the models so that the same description can be given for all of them.

Degradation parameters consist of laboratory half-life, together with the temperature and moisture conditions and the factor Q_{10} - describing the increase in degradation rate with a 10°C increase in temperature. For PRZM2, a field half-life is used with no correction for temperature and moisture. As would be expected, a change in half-life has a considerable impact on total residue levels. Changing the DT_{50} value had a greater effect on PRZM2 than on VARLEACH and LEACHP, both of which gave similar responses. Increasing half-life by a factor of 1.5 resulted in an increase in total residues of 29% and 31% with LEACHP and VARLEACH, respectively; with PRZM2 the increase was 43%. The relationship was not linear, with the rate of increase of total residues with increasing half-life reduced at half-lives of more than 100 days. The other input parameters involved in calculation of degradation rates were of similar importance.

Input parameters describing **adsorption** in soil are K_d or K_{oc}-values, together with a Freundlich coefficient (LEACHP) or a factor for increase of adsorption over time (adsorption increment, VARLEACH). The adsorption increment led to greater retention in the top layers of the soil profile, but its influence was not investigated in detail - on account of its relevance to only one of the models. A decreasing Freundlich coefficient had a slight influence on leaching depth, which was noticeable mainly with the smallest K_d tested (0.5). The change in K_d itself had a very strong influence on leaching depth, as would be expected. There was no difference in the sensitivity of the four models tested, although when K_d-values were very small (< 1.0), LEACHP showed a slightly greater increase in leaching depth than the other models. The relationship was not linear - at K_d-values less than 1.0 the effect of a change was much greater than at larger values. The effect of change in K_d on total residues was very small and did not differ between the models. The total residues increased by about 10%, with an increase of K_d from 1.7 (standard) to 8.0. At the smallest K_d value (0.5), total residues were about 5% less than for the standard value - in all models.

Water solubility of the chemical, required in LEACHP and VARLEACH, had no influence on simulation results at solubilities greater than 5mg/l. Below this, however, there was an extreme influence on leaching depth, and also on total residues in the case of LEACHP. With an increase in solubility from 1 to 5 mg/l, mean leaching depth increased by 70% for VARLEACH and by 110% for LEACHP. Total residues simulated by LEACHP also increased by 70%. This effect will also depend on application rate, since the influence of water solubility in both models is to allow the existence of solid pesticide when the total amount present exceeds the calculated equilibrium distribution between adsorbed and solution phase material.

Soil Parameters

Differences in the soil properties required by each model are much greater than for the

applied chemical. Therefore most of them cannot be compared between all the models, although some have considerable influence on simulation results.
The effect of **bulk density** on total residues (μg/kg soil) - decreasing residues with increasing bulk density - is simply a result of calculations within the model. The amount per volume (in the simulated soil layer) does not change. There was also an effect on leaching depth, which was greatest for VARLEACH. When bulk density was decreased from 1.5 (Standard) to 1.0, mean leaching depth increased by 30% for both LEACHP and PRZM2, and by 57% for VARLEACH. Increasing bulk density to 1.9 gave a decrease in mean leaching depth of 14% and 20% for LEACHP and VARLEACH, respectively.

Variation of **Initial soil moisture content** had little effect on results from any of the models. In many situations, the value of this parameter is not known, but this is unlikely to lead to errors in the simulations.

Field capacity is used in PRZM2 (volumetric) and VARLEACH (gravimetric). In PRZM2, changes in field capacity had little effect on the results. Total residues and the shape of the distribution curve altered little. In VARLEACH, changes in field capacity had a significant effect on total residues, presumably due to the influence of variation in moisture content on degradation. Leaching depth was not affected.

Saturated conductivity is used to model the soil water regime in LEACHP only. Decreasing the value of this parameter to 2 mm/day (Standard: 19 mm/day) led to a slightly shallower leaching due to restricted rainfall infiltration. With an increase above the standard value, the shape of the peak did not change, but water flow out of the soil profile increased, leading to a drier soil and therefore restricted degradation. The loss of pesticide by drainage, however, increased only very slightly from 0.1mg/m^2 (Standard) to 0.4 mg/m^2 (2000 mm/d).

Dispersivity is an important parameter in LEACHP. Numerical dispersion is introduced according to segment thickness, but is then eliminated using an empirical relationship. The user must also specify a value for dispersivity which is used to calculate the hydrodynamic dispersion coefficient and thus introduce analytical dispersion into the model. LEACHP sets a minimum value for dispersivity which is equal to the correction for numerical dispersion. Thus in the present tests increasing dispersivity from 0-10 mm had no effect on simulation results. Above this threshold value, the model is very sensitive to changes in dispersivity. A value of 20 mm for dispersivity resulted in a reduction in peak concentration of approximately 10%, but no increase in drainage loss; whereas drainage losses increased very markedly and the peak in soil residues became very flat with dispersivity values of 50 mm or more. Choice of input parameters is important and, in the experience of the authors, setting dispersivity equal to layer thickness is a good first approximation.

Numerical dispersion is also introduced into PRZM-2 and will vary according to the thickness of the soil segments specified by the user. Thus selection of these thicknesses is very important, particularly for the top few cm of soil, and the guidelines set out in the user manual should be followed as closely as possible. The user is given the opportunity to eliminate numerical dispersion by a methods of characteristics procedure. In this instance, a value for the hydrodynamic dispersion coefficient is required to introduce analytical dispersivity into the model, and this option should only be selected if field data are available for calibration.

In VARLEACH, segment thickness is always set to 1 cm, and this introduces a pre-defined amount of numerical dispersion into the calculations.

Initial soil temperature is used only in LEACHP. It influenced only the total residues remaining in the profile. Use of lower initial temperatures resulted in higher residues, demonstrating that this factor must be defined as accurately as possible in LEACHP.

2.2.7 Modelling with Standard Scenarios

(A Walker, J E Cullington, HRI Wellesbourne, UK & R Calvet, INRA Grignon, France)

Introduction

One of the main advantages of modelling is the ability to make generalised predictions of pesticide fate and to investigate the likely variability in behaviour in a much wider range of situations than is possible experimentally. Several computer models were used to estimate the influence of variability in weather parameters on potential leachability of a test pesticide, and to estimate the likely influence of variations in pesticide properties on potential leachability under standard circumstances. Much of the work was done with the VARLEACH model, although comparative calculations were also made with LEACHP and to a lesser extent PRZM2. Further calculations were made with the simple screening model CMLS in order to test its predictive ability relative to that of the other more sophisticated models.

2.2.7.1. VARLEACH, LEACHP and PRZM2 - Sensitivity to weather data

Weather data sufficient to run the various models were available from a sequence of years from sites in France, Germany, Italy and the UK. These data were used with three standard soil profiles selected as examples of major soil types within Europe in which significant leaching might be expected. Properties of the profiles are listed in Table 2.4. The VARLEACH model was run for successive 2-year periods beginning on January 1st of different years, with input parameters to describe the different profiles and with pesticide data representative of the herbicide atrazine. The results are summarised in Figure 2.12 in which the amount (g ha^{-1}) predicted to leach below a depth of 60 cm following application on January 1st of the different years is shown for each site and each weather data set. The results from the simulations illustrate wide variation between years and between sites in the predicted leachability of atrazine. In the sandy loam (profile 1), the amount calculated to leach below 60 cm varied from 1.5 g following application on January 1st 1990 to over 30 g (3% of the applied dose) following application on January 1st 1979. Similar relative differences were predicted for the other standard profiles. Least leaching was predicted for the luvisol which had

Table 2.4 Soil properties

Profile 1 (UK, Cambrisol)

Horizon (cm)	0-20	20-40	40-60
Sand (%)	81.0	83.0	85.0
Silt (%)	10.0	12.0	12.0
Clay (%)	9.0	5.0	3.0
OC (%)	1.0	0.8	0.3
FC (%w)	12.1	10.2	8.4
Bulk density	1.4	1.4	1.4
T½	40.0	60.0	84.0
Kd (atrazine)	1.0	0.8	0.3

Profile 2 (France, luvisol)

Horizon (cm)	0-20	20-90	>90
Sand (%)	5.0	15.0	40.0
Silt (%)	80.0	55.0	40.0
Clay (%)	15.0	30.0	20.0
OC (%)	1.3	0.3	0.1
FC (%w)	23.5	34.5	21.9
Bulk density	1.3	1.3	1.3
T½	40.0	185.0	1000.0
Kd (atrazine)	1.3	0.3	0.1

Profile 3 (France, podzol)

Horizon (cm)	0-15	15-40	>40
Sand (%)	60.0	85.0	20.0
Silt (%)	30.0	10.0	30.0
Clay (%)	10.0	5.0	50.0
OC (%)	1.5	0.4	0.7
FC (%w)	11.8	5.9	28.3
Bulk density	1.5	1.5	1.5
T½	40.0	240.0	600.0
Kd (atrazine)	1.5	0.4	0.7

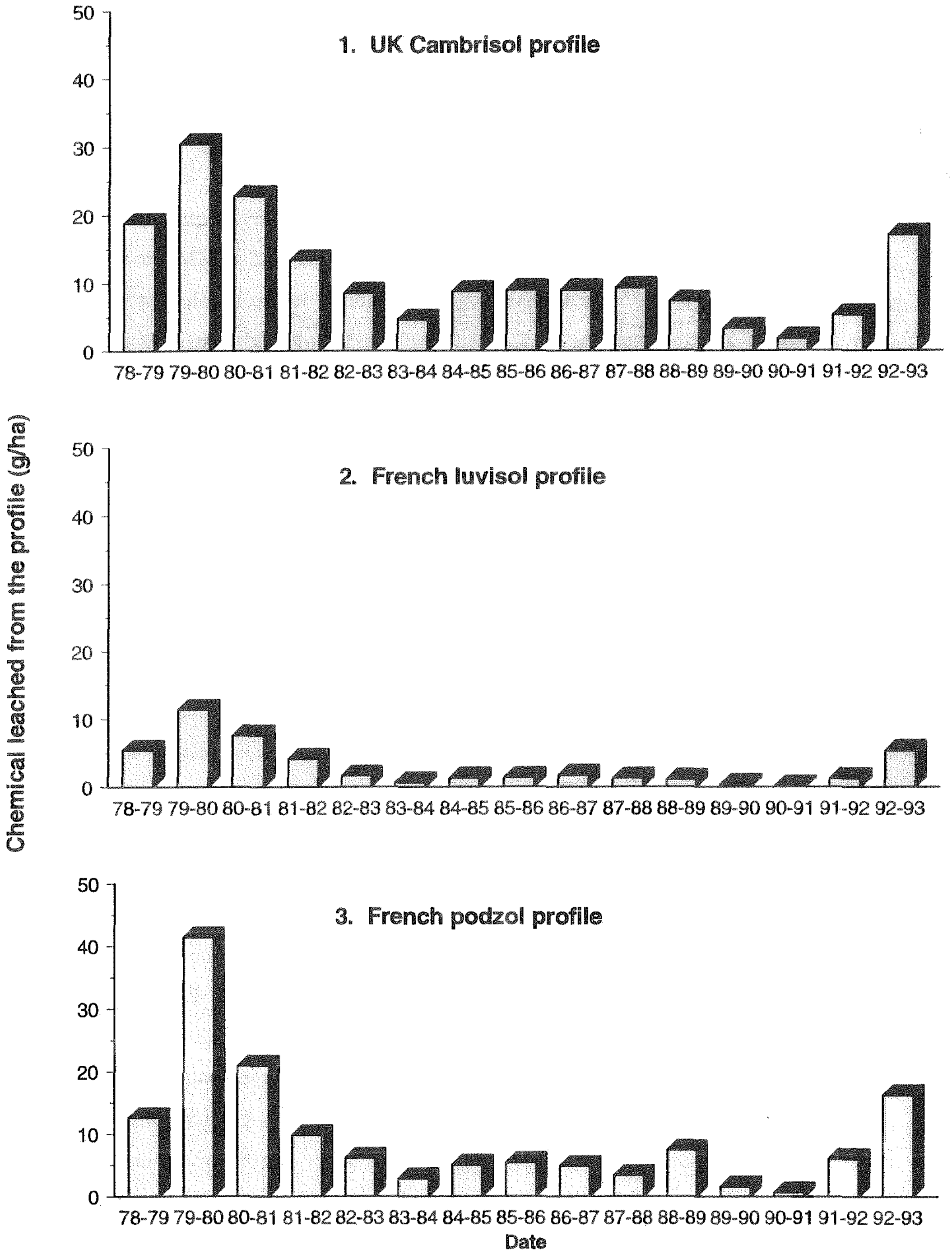


Figure 2.12 VARLEACH calculations of atrazine leaching from three soil profiles using weather

more organic matter and much larger water holding capacity than the sandy loam. The podzol profile had the largest organic content of the three test soils, but also held the least amount of water at a given potential. These low water holding properties may well explain the very high leaching predicted in the wettest seasons (1979/80). The modelling exercise was repeated for the sandy loam profile only, using the LEACHP and PRZM2 computer models. The results are shown in Figure 2.13. The results indicate a similar order of predicted leachability between years, but suggest somewhat smaller overall fluctuations in the data predicted by LEACHP and PRZM2 than those predicted by VARLEACH. The relatively small variability in the PRZM2 predictions can probably be explained by the model assuming identical degradation rates in all years and only allowing for differences in leaching behaviour. In both of the other models, the differences in weather patterns will influence degradation as well as mobility. The results strongly suggest that it is important to take account of both factors. Further comparative calculations were made using VARLEACH and atrazine in the sandy loam profile using weather data from various European sites. The results are shown in Figure 2.14. In general greatest potential leaching was predicted using weather data from Wellesbourne, England. Least leaching was predicted using the weather data from Berlin and Paris, although the weather data sets were relatively restricted. The greatest variability was recorded using weather data from Beziers, but strict comparisons are again difficult because of the very wide data base available from this site. Within the 60 model runs, the maximum predicted leaching losses account for less than 5% of the initial dose. Attempts were made to relate the predicted losses shown in Figure 2.14 to various parameters associated with rainfall, evaporation, and predicted drainage. Some of the observed relationships are shown in Figures 2.15. These results illustrate a similar relationship between drainage and atrazine losses at the three north European sites, but very different relationships at the sites in Italy and Southern France. The much higher temperatures in Italy presumably result in more rapid degradation and hence small leaching losses; at the site in Southern France, the great variation in the predicted losses probably results from the highly irregular and unpredictable rainfall patterns. At this latter site, it was not unusual for as much as 10% of the annual rainfall to occur in just one day.

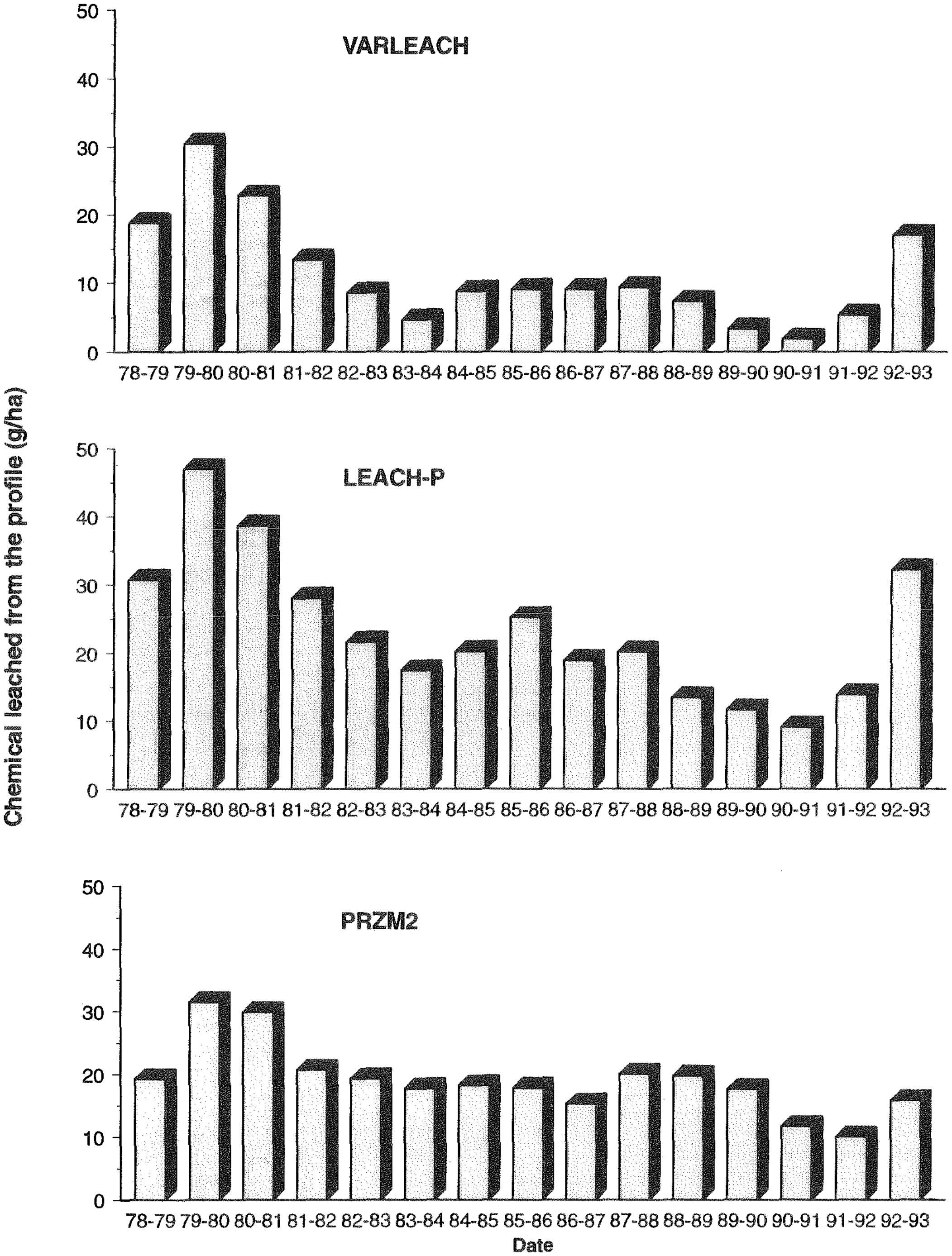


Figure 2.13 VARLEACH, LEACHP and PRZM2 calculations of atrazine leaching from a sandy loam profile using weather data from Wellesbourne, 1978-93

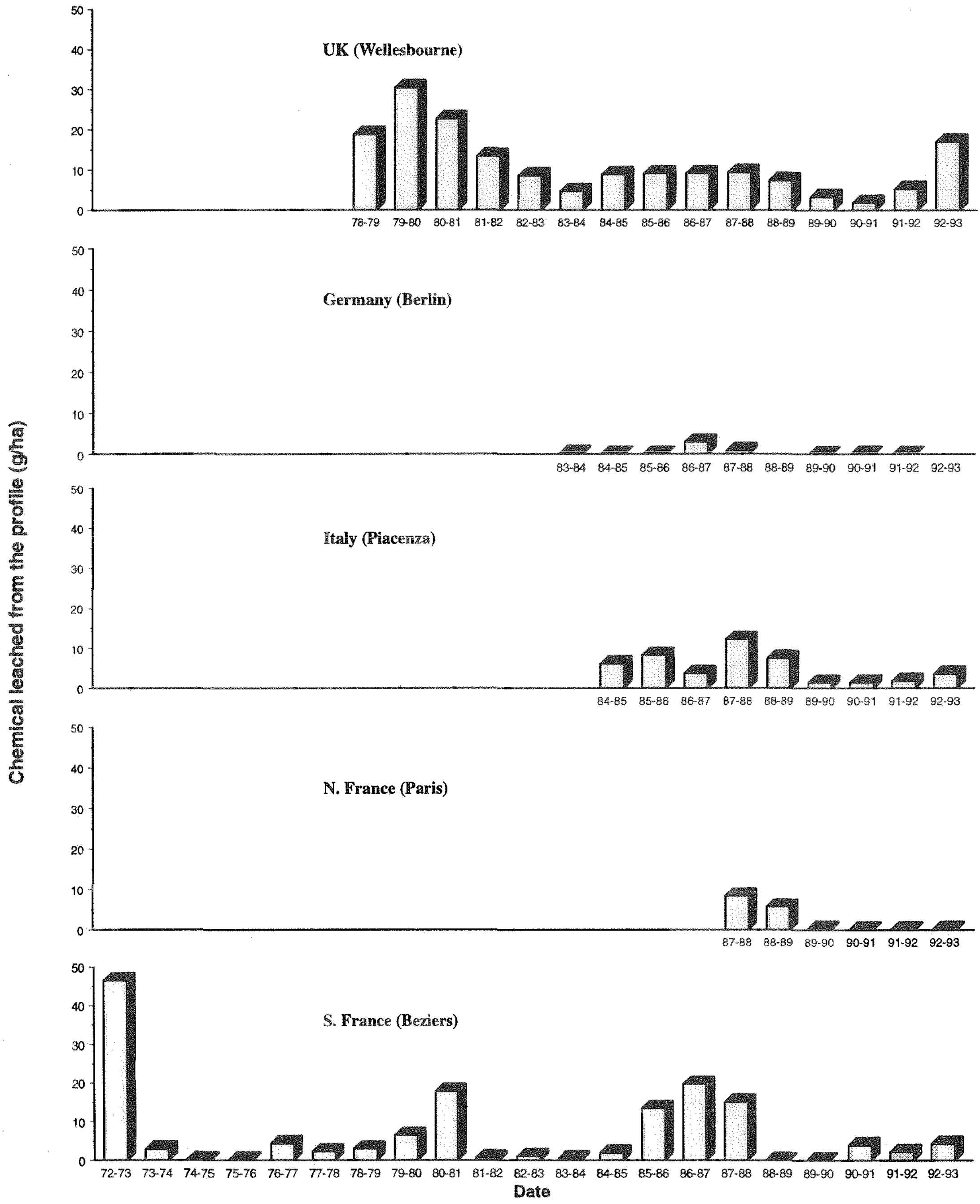


Figure 2.14 VARLEACH predictions of atrazine leaching from a sandy loam profile using weather data from five European sites

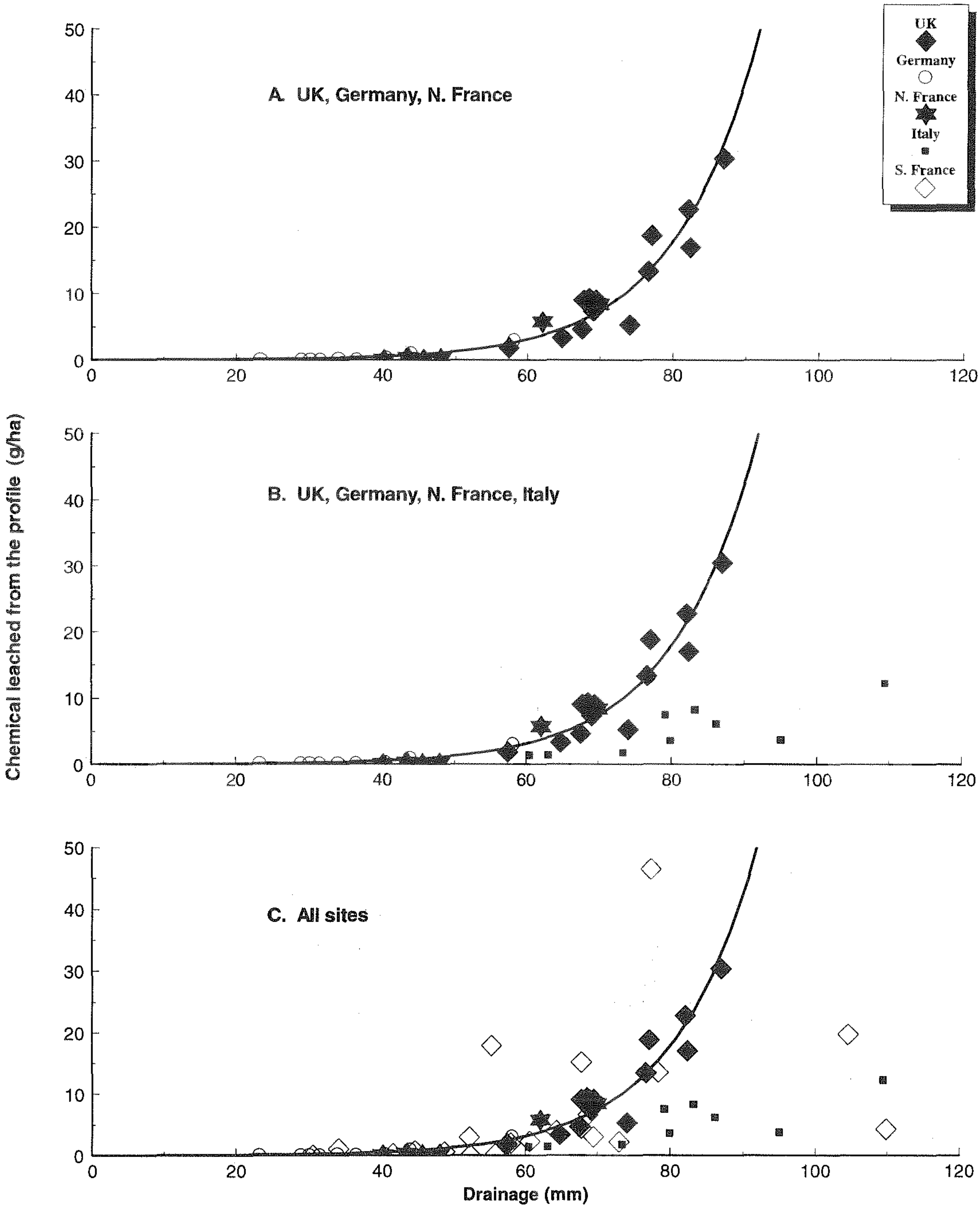


Figure 2.15 Relationship between VARLEACH predictions of Atrazine leaching from a sandy loam profile and volumes of drainage water

2.2.7.2 VARLEACH, LEACHP and PRZM2 - Sensitivity to pesticide properties

The weather data from the United Kingdom for the 2-year period showing the greatest potential leaching were used with the appropriate input parameters to characterise profile 1 (the sandy loam profile, Table 2.4) in a sequence of calculations to investigate the influence of pesticide properties on potential leachability. The pesticide properties used were values of Koc in the range from 10 to 800 in combination with half-lives (at 20°C) of 10, 20, 35, 50, 70 or 90 days. The results obtained with the VARLEACH model, expressed as before as g/ha leached below 60 cm are shown in Figure 2.16 and they indicate a range up to a maximum of about 950 g/ha. The model also calculates the total volume of water passing through the lower boundary of the soil and hence it is possible to derive the average concentration in the water that has moved through the upper soil profile. Because of differences in travel time, this overall leachate concentration was calculated using the total flow during the maximum period of the simulations (i.e. 2 years). The results are shown by the right hand axis in Figure 2.17 and the range of 0.1 to 1000 g/ha (left hand axis) converts to average leachate concentrations of 0.012 to 125 μg/litre (right hand axis). Individual compounds can be positioned within the body of this diagram according to their respective half-lives (20°C) and Koc values taken from the literature, and some examples are shown in Figure 2.17. The vertical position of a chemical within the diagram therefore indicates its potential to contaminate deeper soil layers relative to that of other compounds, and gives a quantitative prediction of pollution potential. The data in Figure 2.17 are presented in an alternative format in Figure 2.18 to illustrate further the predicted relative pollution potential of the different compounds. A detailed discussion of how this modelling approach integrates with other systems for assessing the leaching potential of pesticides was presented by Walker & Hollis (1994).

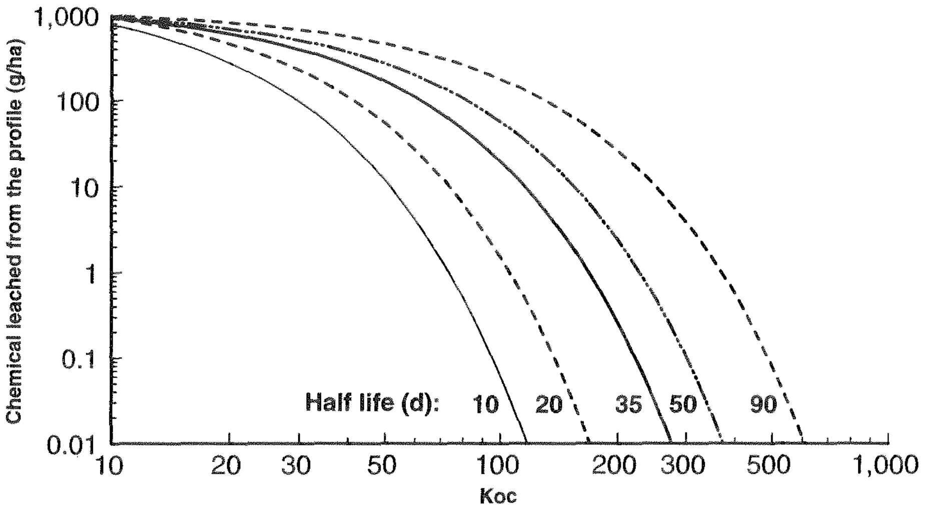


Figure 2.16 **Influence of Koc and half-life on VARLEACH predictions of potential leachability from a sandy loam profile using weather data from Wellesbourne, UK, 1979-1980**

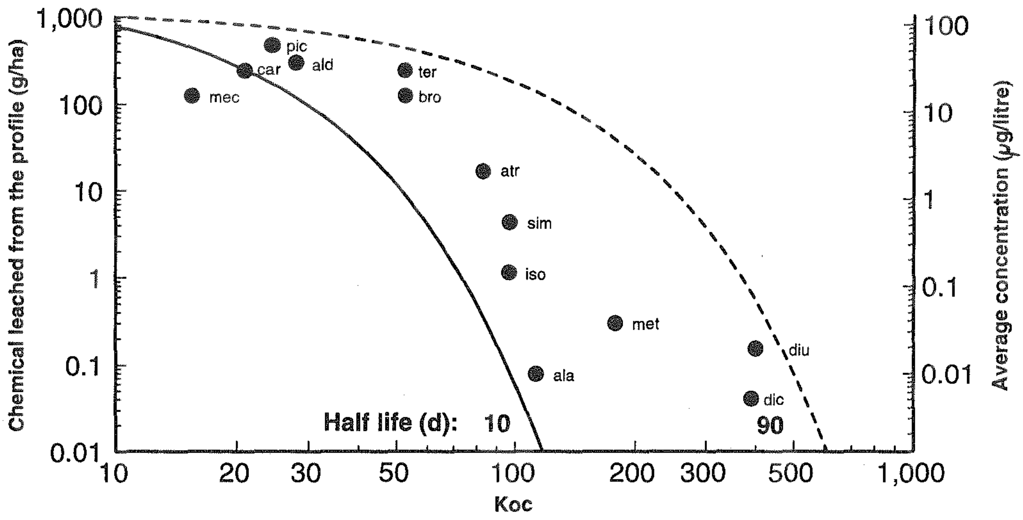


Figure 2.17 **VARLEACH predictions of relative leaching potential of alachlor, aldicarb, atrazine, bromacil, carbofuran, dichlobenil, diuron, isoproturon, mecoprop, metolachlor, picloram, simazine and terbacil (based on Figure 2.16)**

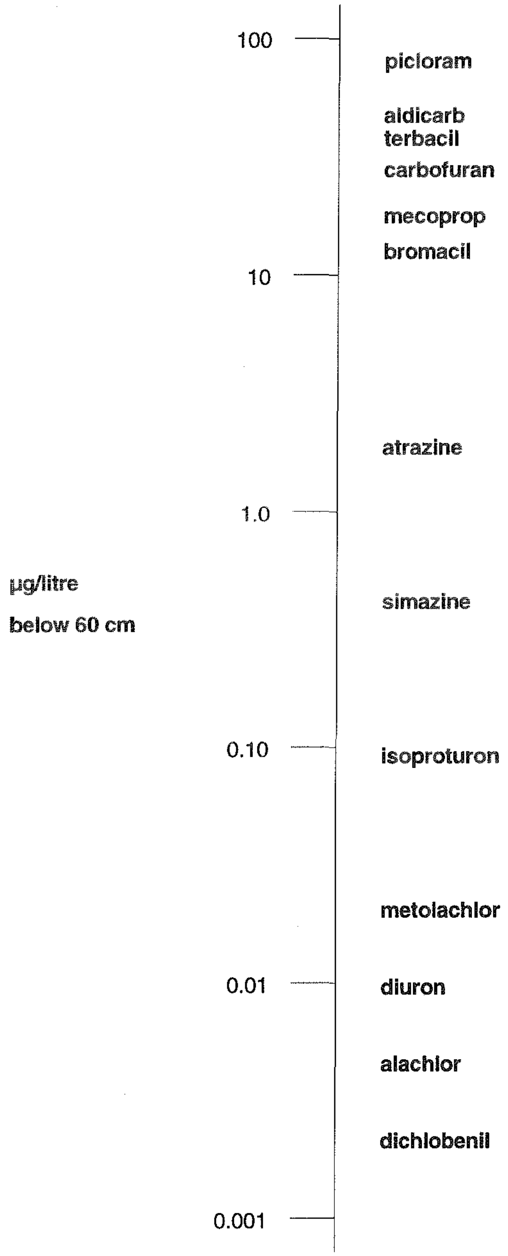


Figure 2.18 VARLEACH predictions of relative leaching potential of alachlor, aldicarb, atrazine, bromacil, carbofuran, dichlobenil, diuron, isoproturon, mecoprop, metolachlor, picloram, simazine and terbacil (based on Figure 2.16)

2.2.7.3 Assessing groundwater pollution potential with CMLS

The model CMLS (Continuous Modelling in Layered Soils) describes the movement and degradation of a soil-applied chemical. For any given time scale, it can be used to calculate the residual amount of a chemical in the profile and the depth reached by the solute. The required input data are:

- Soil characteristics for each horizon: thickness, organic carbon content, bulk density, moisture content at applied pressures of 0, 100 and 1500 kPa
- Climatic data: Rainfall and evaporation
- Chemical characteristics: Koc and half-life
- Rooting depth of crop and pesticide application date

Soil and chemical parameters are allowed to vary with depth, but half-life is independent of temperature and soil moisture content. Adsorption is characterised as an equilibrium partition coefficient. A capacity algorithm is used to simulate soil moisture movement.

A ground water pollution index (GWPI) has been tentatively defined using simulated results with the CMLS model as :

GWPI = solute depth x remaining amount, for a given time

This definition uses the fact that the model calculates the solute transport such that all the solute is localized at the maximum leaching depth. The solute depth is expressed in cm and the pesticide residue as a percentage of applied amount. For a given soil and weather data set, it is possible to calculate this index for several molecules and to classify them according to their GWPI values. The greater the GWPI, the greater is the pollution potential.

The first step in this approach was to calculate GWPI values for:

Case 1: Two soils and a given set of climatic data; the soils were a podzol (POD) and a "sol brun lessivé" (luvisol, BRL), the profiles of which were described in Table 2.4. The weather data used were from an experimental site at Rambouillet for 1990-91.

Case 2: Two sets of climatic data and a given soil; the soil was a podzol (Table 2.4) and the weather data were those of Rambouillet 1990-91 and those of Mont St. Michel 1983-84. These weather data sets were chosen because they were very different: **Ramb90**: Rambouillet 01-04-1990 to 31-03-1991; total precipitation, 597 mm; total evapotranspiration, 776 mm. **R8384**: Mont St. Michel 01-04-1983 to 31-03-1984: Total precipitation, 928 mm; total evapotranspiration, 662 mm.

GWPI values were calculated at 300 days as a function of Koc with half-lives of 25, 50 or 200 days. The calculated values of GWPI are shown in Figures 2.19A and 2.19B for Cases 1 and 2 respectively. The results indicate that:

- molecules can be classified according to their relative groundwater pollution potential if it is assumed that such values at t=300 days are a good criterion (which remains to be verified).

- Variations of both soil and climatic factors can be taken into account in this simple index.

- There is a greater soil and climatic influence if Koc is less than 100 and when half-life is less than 200 days.

This simple approach requires further evaluation to determine its possible usefulness as a technique for prediction of pollution potential.

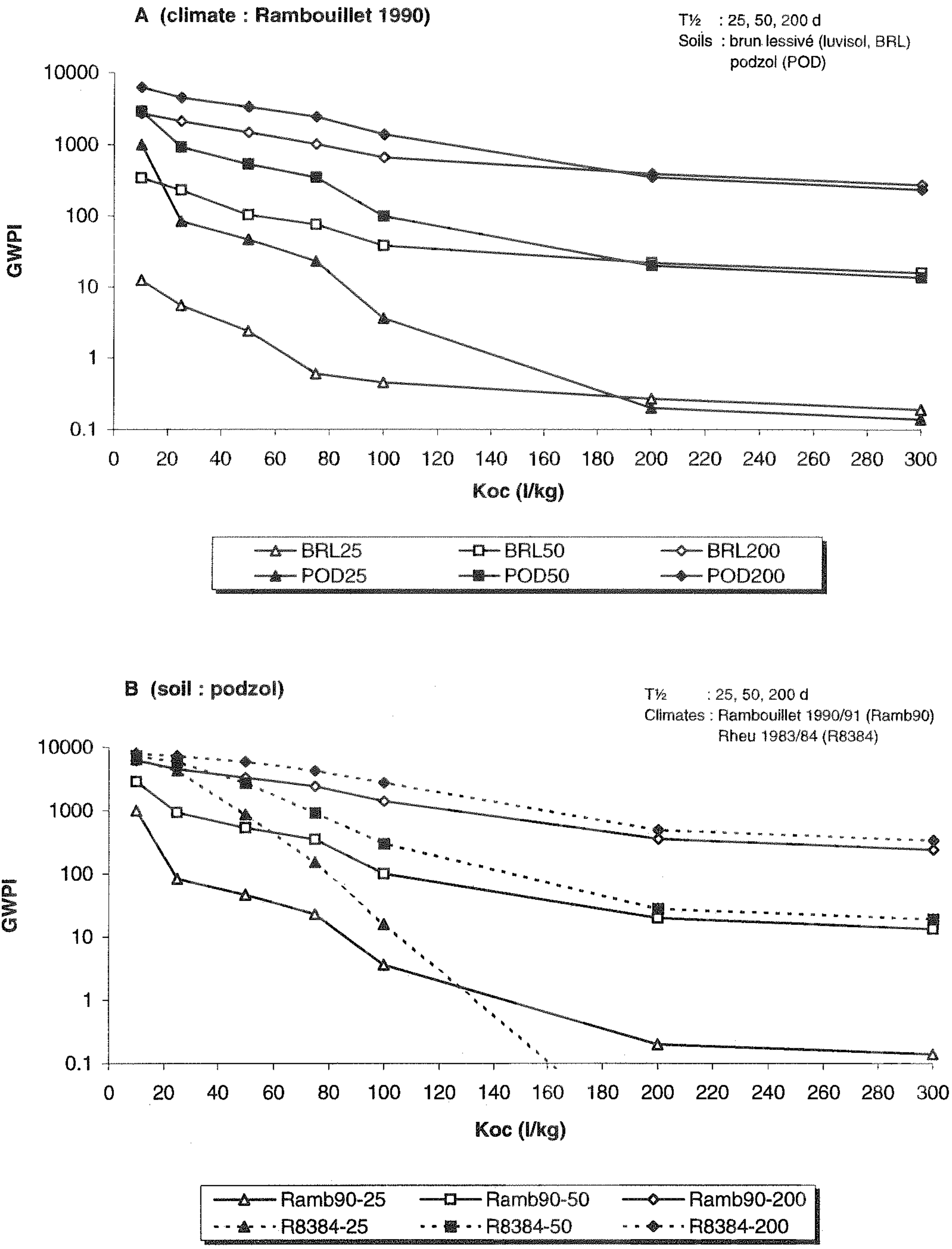


Figure 2.19 Groundwater pollution index (GWPI) calculated using the model CMLS with data to describe two soil profiles (A) and weather data from two French sites (B)

2.2.8 Ring Testing

(C D Brown, SSLRC Silsoe, UK)

To date, no modelling studies have been carried out to investigate the reproducibility of output results from a given model when used predictively by a range of workers to simulate the same scenario. An inter-laboratory ring test was thus carried out by all five research groups. Data from a field experiment carried out in the UK were summarised according to the standard format developed for the experimental database and circulated without giving any results from the study.

The experiment to be simulated involved autumn application of a novel fungicide to a winter wheat crop grown on a sandy loam soil and monitoring for movement to 1-m depth with a series of 9 suction lysimeters over a period of 220 d. At the end of the experiment, soil samples were taken to a depth of 1 m with 10-cm increments at three randomly-selected locations. Modellers were given all available information for selection of model input parameters, including details of pesticide, crop, soil and weather. Each group was asked to predict pesticide concentrations in soil water at each sampling interval, and the distribution of residues in the soil profile at the end of the experiment using LEACHP, PRZM2 and VARLEACH. All modellers were experienced in the use of the three models tested, although most users were least familiar with PRZM2.

The results of the ring test are summarised in Table 2.5. There was a consistent difference in the ability of the three models to simulate the observed leaching of pesticide, which was reflected in predictions of the maximum concentration of pesticide at 1-m depth and the maximum depth with soil residues >0.5 μg/kg. For leaching, goodness-of-fit decreased in the order LEACHP > PRZM2 > VARLEACH. Total pesticide residues remaining in the soil at the end of the experiment were poorly simulated by LEACHP; this aspect of pesticide behaviour was best predicted by PRZM-2, with VARLEACH intermediate in performance.

Table 2.5 Summary of observed and predicted results for the modelling ring test

Output	Observed value	Predicted value				
		1	2	3	4	5
Maximum pesticide conc. at 1 m depth (μg/l)	0.57					
LEACHP		0.22	0.14	0.04	0.16	0.22
PRZM-2		0.02	0	0.11	0	0.01
VARLEACH		0	0	0	0	0
Max. soil depth with residues >0.5 μg/kg (cm)	70-80 cm					
LEACHP		50-60	50-60	30-40	50-60	70-80
PRZM-2		40-50	30-40	60-70	20-30	30-40
VARLEACH		20-30	20-30	20-30	20-30	20-30
Total pesticide residue in soil after 220 d (g/ha)	78.5					
LEACHP		48.3	75.2	16.4	105.3	122.6
PRZM-2		66.8	59.3	63.2	72.0	71.1
VARLEACH		70.5	66.0	82.5	50.3	115.8

Although all modellers received the same information from which to select input parameters for modelling, no two simulations with any of the three models were identical even for the relatively simple VARLEACH model. This resulted from the requirement for a number of key input parameters which could not be derived from the experimental information provided, and were thus open to considerable subjectivity - according to the experience and knowledge of the individual modeller. The main subjective parameters identified were dispersivity in soil, initial soil conditions such as temperature and moisture content, and unquantified pesticide parameters such as factors describing changes in rate of degradation with changes in soil temperature or moisture content and physico-chemical factors such as the air diffusion coefficient. For example, in simulation 3, LEACHP predicted considerably more pesticide degradation than in the other four simulations. This resulted from use of a literature-derived value for the average diffusion coefficient of pesticides in air, which was 50% larger than the default value supplied with the model. This resulted in an unrealistic level of pesticide volatilisation. However, no indication of the sensitivity of the model to changes in the air diffusion coefficient was given in the manual, even though this parameter is seldom measured. Similarly, the large differences in predicted leaching of the pesticide between the five simulations with PRZM-2 were mainly the result of subjective selection of segment thickness for modelling, which controls the amount of dispersivity in soil and hence the degree of leaching. It is important to note that the parameters identified as introducing subjectivity are seldom available as site-specific measurements for modelling - except in intensive, specifically-designed experiments. Subjective selection of their values is thus the rule rather than the exception.

Differences between the five leaching predictions for each model decreased in the order PRZM-2 > LEACHP > VARLEACH and this is matched by a decrease in the number of parameters which may be used for simulations (118, 81, and 20, respectively). The time series for observed and predicted pesticide concentrations in soil water are shown in Figure 2.20 for the five simulations carried out with LEACHP, along with the standard deviation associated with both observed and predicted results. Timing of breakthrough of pesticide was well predicted by LEACHP, but the magnitude of the peak concentration was underestimated and its timing was delayed compared to observed results. Generally, the standard deviation for the five simulations with LEACHP was smaller than the discrepancy between observed and predicted data. By the end of the experiment, observed and predicted concentrations of pesticide were very similar and the standard deviations associated with measured and simulated concentrations were of a similar order of magnitude.

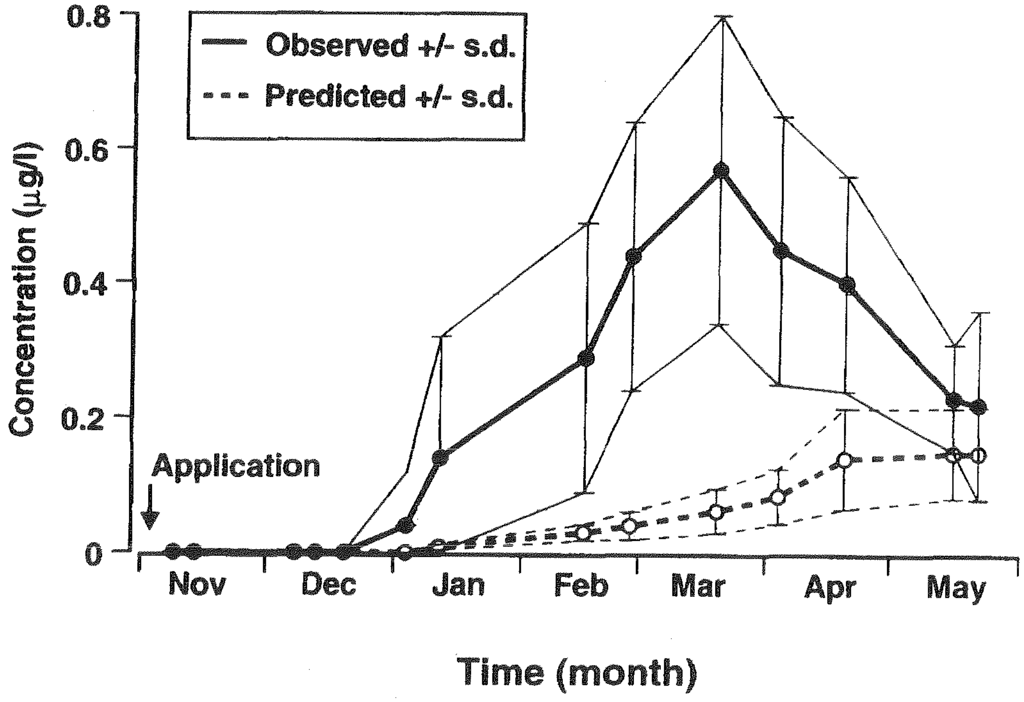


Figure 2.20 Comparison of observed concentrations of pesticide in soil water at 1-m depth (mean of nine samples ± one standard deviation) with LEACHP predictions (mean of five simulations ± one standard deviation)

The ring test shows that modelling results for the same scenario can vary between users by amounts similar to the variation in pesticide measurements made in the field. This user-dependence has not been previously considered, but should be an important component of evaluation of any model output. Model developers should be encouraged to reduce model subjectivity by decreasing the number of parameters which are not readily measured and providing detailed guidance for selection of such parameters that cannot be eliminated. Model output needs to be accompanied by a detailed list of the parameters used for simulations along with a brief description of the rationale for parameter selection.

2.3 MODULAR ANALYSIS AND RESTRUCTURING OF MODELS

(A A M Del Re, M Trevisan, UCSC Piacenza, & E Brasa, Microring, Milan, Italy)

2.3.1 History

The objective of this part of the project was to develop a flexible and updatable structure able to act as a link between model developers and users, in order to allow the latter to choose from the many different options currently available. This is important since no one model can be considered as universally valid for all contexts. Furthermore almost all models are subject to continuous updating, with new versions often poorly documented, both in terms of manuals and codes. The structure must be capable of dealing with different models simultaneously, creating new *ad hoc* pathways decided by the user, and operating with new or upgraded models or routines. In addition, the source code must be robust and the program structure well documented. Three models have been selected for structure assessment amongst those most widely used in Europe, namely *VARLEACH, PRZM2* and *LEACHP*. These cover the range of model types presently available for different purposes, in particular research and legislation.

The first step was to analyse the program structures. The three models have a common structure, with a main program which calls single sub-routines sequentially within a basic time loop and also reads the data sets. The main program then performs preliminary calculations in the year loop, day loop and ultimately in fractions of the day loop. Sub-routines accept input from preceding ones and in turn offer their output to other sub-routines. Sub-routines can operate at different temporal scales within the basic time domain.

In view of this common structure, discrete program blocks have been created, maintaining where possible the original routines. This has proved to be relatively easy for PRZM2, less so for LEACHP (in particular for the main program), and has necessarily been carried out practically from scratch for VARLEACH since it was not originally divided into sub-routines. Within each routine, all variables have been classed separately as input or output. These have been organized into a flow (or pipeline) chart, showing the pathway taken by each variable within the program.

2.3.2 Sintax

Specific software was developed to perform the above operations and was given the name *Sintax*. The first version of the program (Sintax 1.0) analyses a single code-block and classifies variables as input/output according to the following criteria:

- Variables which are used before being assigned are classed as "input" ie. they appear on the right-hand side of the equation before appearing on the left.
- Variables which are assigned before being used are classed as "output" ie. they appear on the left-hand side of the equation before appearing on the right.

Local variables have been excluded from the flow chart as they play no part in determining the overall model structure. Classification of variables used as function arguments in code-blocks containing sub-routine calls can be difficult. When using the first version of the program, these had to be classified manually, although such variables did not occur too frequently in the programs analysed. A second version of the program (Sintax 2.0) has been developed under another EU project for a similar study of the *GLEAMS* model. This improved version of the software can perform the described analyses automatically, solving recurrent subroutine calls at any depth level.

2.3.3 Simul

In the first phase of model analysis, *Sintax* classified the role of each variable, distinguishing between local and global variables. Particular attention was paid to global variables, which can be considered as a pool of entities, with each having a unique physical meaning and most with names that clearly describe their role. Each of the three leaching models was then divided into blocks of code (usually sub-routine calls), with each block acting on one of the variables within the global pool. One or more variables from the pool must be present in each block, so that its value, or that of other variables in the pool, can be transformed. Each block, or sub-routine, acts on a small set of variables leading in and out of the pool, with the latter supplying values to other blocks (Figure 2.21).

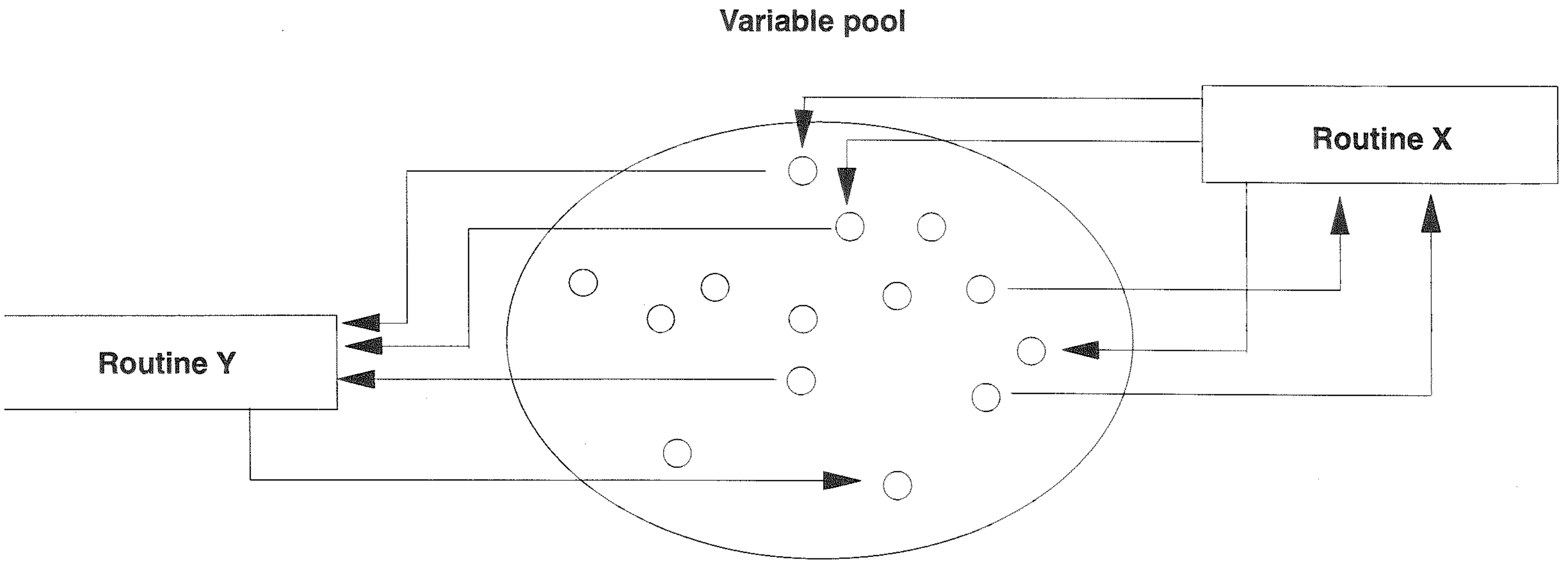


Figure 2.21 Structure and function of Simul

A sub-routine which has input variables set by another code block reflects a relationship (determined by the execution time sequence) which can be visualised in a schematic diagram (Figure 2.22). Such diagrams were useful in clarifying the structure of each program. This was further achieved by formulation of a "precedence scheme", describing relationships between variables by means of flow charts. These schemes are complex and difficult to manage. With this in mind, software was developed, termed *Simul*, which simulates the flow of information in each model during execution and describes the route taken by each variable. Calculations within each model are not actually performed, but global variables updated after execution of each code-block are visualised by means of highlighting. *Simul* also recognises alternative paths of block execution, highlighting “executed” and “executable” sub-routines in different colours. Each model is analyzed as a separate entity, since *Simul* is not able to identify similarities in variable names and roles in the program structure.

2.3.4 The DataBase

The next step was to formulate a unifying overview of the three models. Much of the work entailed rationalisation of variable names and identifying which have similar roles in each of the three models. Each identified variable was classified as a universal variable, and assigned a universal name reflecting its function and logical and non-logical relationship, in addition to its original identity in each program. The overall objective was to establish a basic structure common to all models which, by organising the information available in analysed models, acts as the first step in development of a new model.

Due to the large number of variables in the three models (almost 1000) a relational database was used to store the following information:

- routines or code-blocks in the programs
- variable names in each program
- variable roles in each code block
- universal names assigned to each variable

The graphical layout used to plan the database tables was termed an *Entity-Relations Diagram* (Figure 2.23). Connections between entities are of two types, namely "one to many" and "many to many". For example, in the former, many sub-routines may be present in the same program, but only one program contains a code block. For the latter, many variables may be present in one sub-routine, and there may also be many routines using each variable. The data tables implementing the relationships summarised in Figure 2.23 are illustrated in Figure 2.24, together with their field names.

The database has a number of advantages:

1. storage of large amounts of paper is avoided
2. data can be retrieved rapidly and in a customized way

   associations between the universal names of variables and each model are readily determined
3. flow diagrams describing all variable-routine connections can be traced automatically, using the new software *SimulGraph*, a graphical development of *Simul*.

SimulGraph can draw flow diagrams or pipeline graphs automatically for any set of selected routines, using both the original and the universal names of each variable. An interactive version of *Simulgraph* is currently under development.

2.3.5 The final product: the *Engine*

2.3.5.1 Aims of the product

This software has been written as the basis for development of a unified program. It was not possible to complete this task during the course of this 2-year EU project, and the work will be continued over the next few years.

The *Engine* comprises three main structures:

- the sub-routines (Micromodules)
- the **job-scheduler**

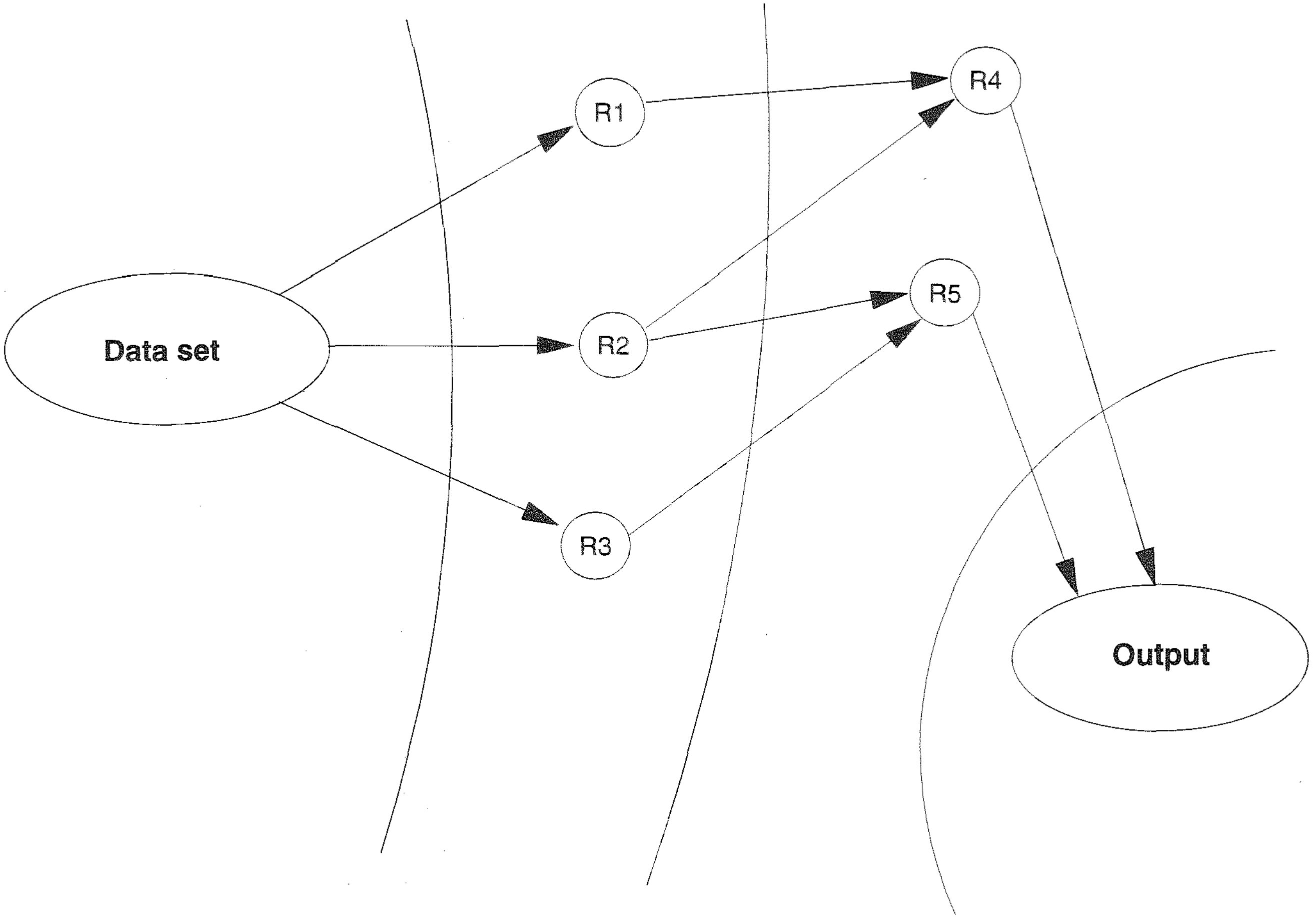


Figure 2.22 Schematic diagram produced by Simul

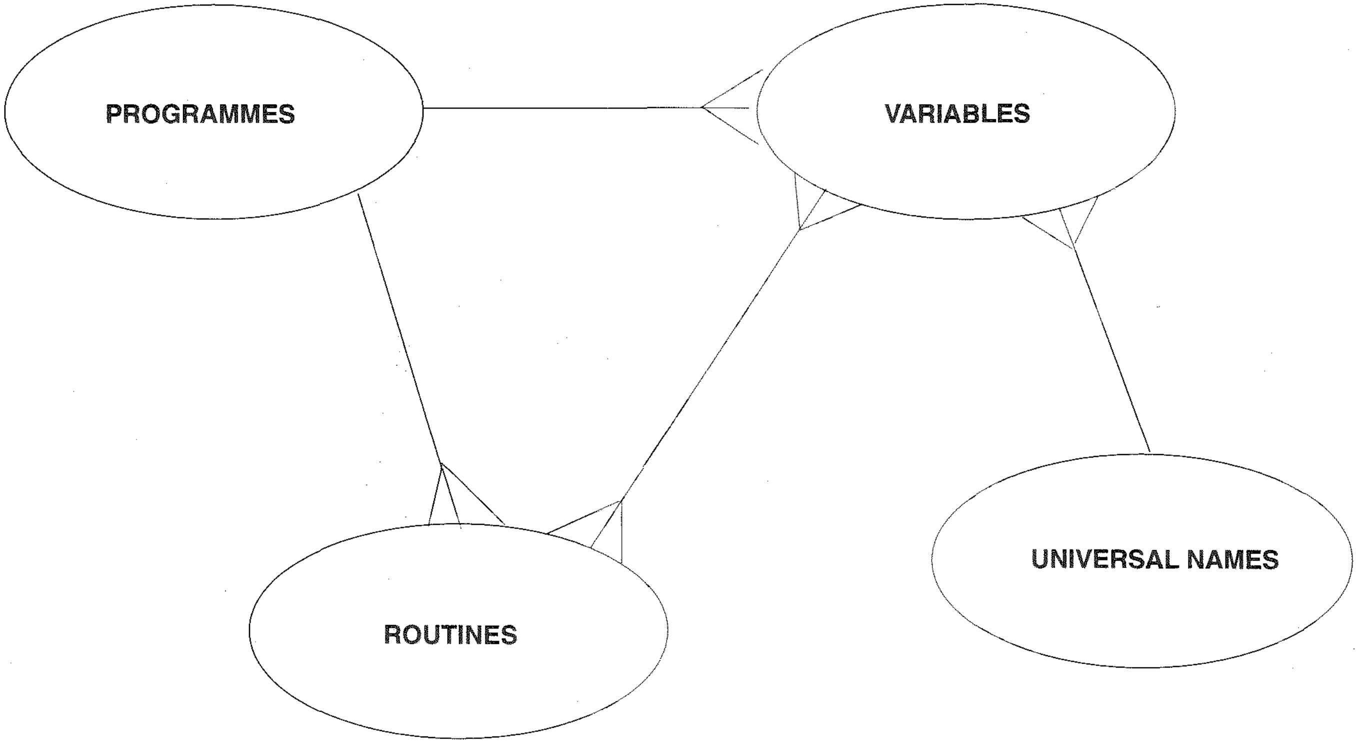


Figure 2.23 Entity relations diagram

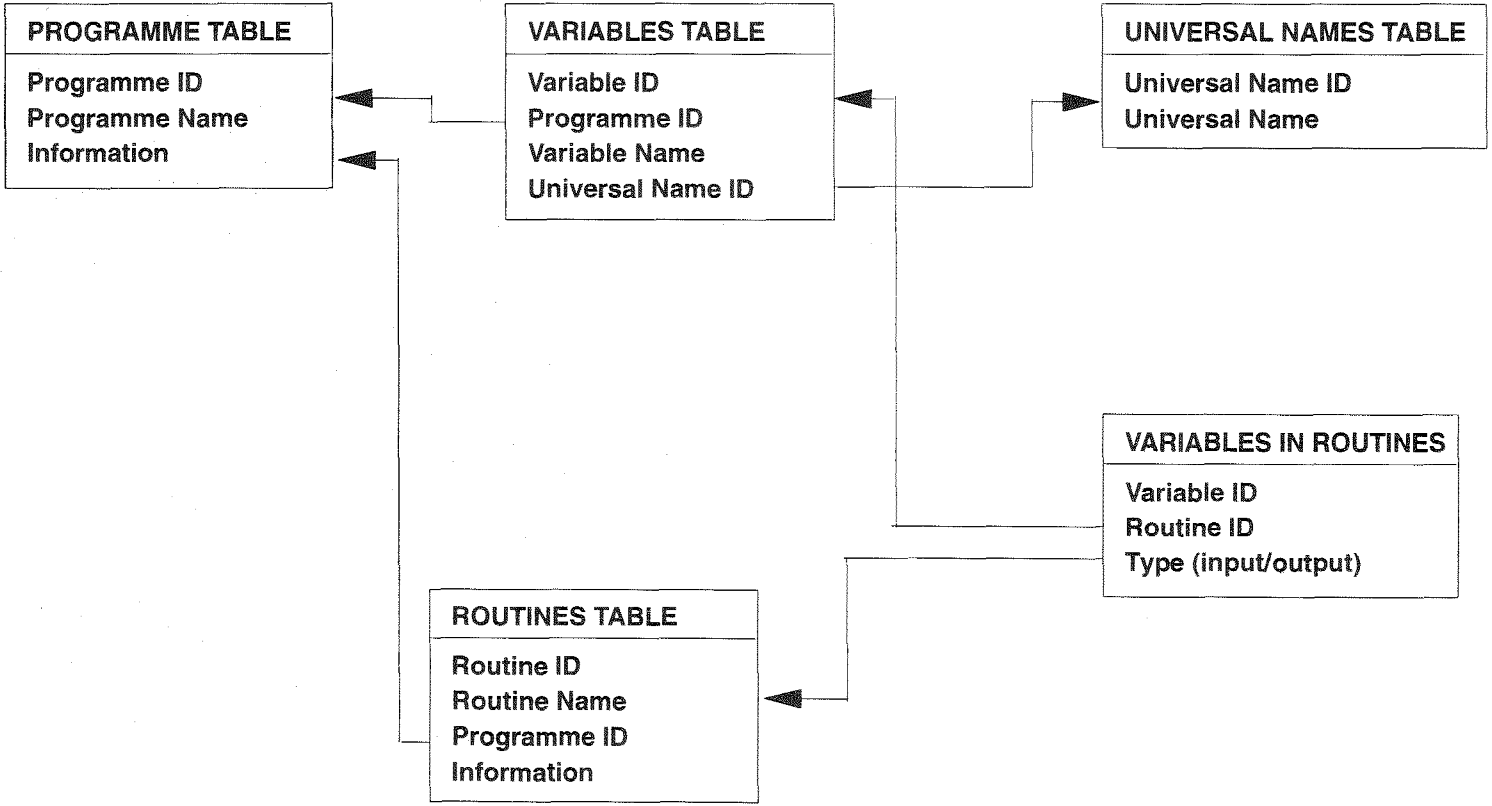


Figure 2.24 Data-base structure used by SimulGraph

- a container for the variables (pools) shared by the Engine and the sub-routines

1. **The sub-routines**

Sub-routines have been identified in the simulation programs, as explained in Section 2.3.1. Where necessary, the code has been split into units corresponding to simple physical processes. These code blocks are termed *micromodules*. In order for the Engine to function, it requires micromodule code and a continuously updated list of those micromodules that have been used.

2. **The job-scheduler**

The main program is essentially a job-scheduler for the micromodules. It is driven by an internal clock and uses the micromodule execution times specified in the scheduling table.

3. **The pools**

Variable pools are similar to those described in Section 2.3.3. All of the variables have been classified into a number of "thematic" pools (see Appendix II). The pools have been provided with specific functions (integer, float, double, array) which are the only way in which variable values in each pool can be read or written to. The pools also contain procedures for reading input data and writing output data. In the current version, inputs and outputs are only by means of computer files; in future versions, a user-friendly interface would be beneficial.

2.3.5.2 The Engine

In the current version, all three structures (sub-routines; job-scheduler; variable pools) are compiled together. The job-scheduler reads the scheduling table and runs the micromodules at the specified times. At its run time, each micromodule accesses the required variables from the pools, or from more global variable values. Each module, when run, can update the scheduling table. It is therefore possible for micromodules to be dedicated to flow control.

The *Engine* is written in C^{++} language, for the reasons outlined in Appendix II, and a more detailed discussion of the structure of the *Engine* (with some examples of code) is also given in Appendix II.

2.3.5.3 Stage of development

The general structure of the Engine has been written and tested with micromodules extracted from the VARLEACH and LEACHP programs. The VARLEACH code has been fully separated into micromodules and implemented in the new structure. Time loops (the main daily loop plus the two day-fraction loops for calculation of evaporation and degradation) are driven by the job-scheduler. The boundary loop is driven by a programmed loop inside a sub-routine. Two water-flow sub-routines have been implemented - one from VARLEACH and one from LEACHP, and these have been used to provide alternative paths of simulation. Some of the difficulties encountered when micromodules from one program are substituted by micromodules from another are discussed in Appendix II.

2.3.5.4 Advantages of the *Engine* structure

The *Engine* structure confers many benefits to model development by means of its characteristics:

-**Simplicity**: Achieved by making sub-routines as small as possible and corresponding to physical processes in the model. Variables are given common names, for instance SoilWaterContent, SimulationStartingDate etc.

- **Completeness**: Providing the capability to study all possible simulation strategies based on the existing models, by virtue of the pipeline diagrams based on universal names.

- **Upgradeability**: Achieved by means of many sophisiticated software analytical "tools", and by segregation of "main parts" and code-blocks. The resulting program is therefore highly modular, a direct result of the analytical style adopted at the start of the project. This permits rapid analysis of other simulation programs written in other procedural languages (such as Fortran, C, Pascal, Basic etc). When perfected, this technique of micromodule programming and scheduling will permit any new subroutine to be easily incorporated into an existing program.

3. GENERAL DISCUSSION AND CONCLUSIONS

3.1 GENERAL COMMENTS ON MODEL USE

3.1.1 Ease of Use

Models can be used to extend basic knowledge, to manage the output data from scientific experiments, to aid in the preservation of environmental resources, as a management tool, and to aid legislative procedures. For each of these uses, a different type of model may be needed, and of those available, not all may be equally reliable. For each use, different selection criteria may apply. The input data requirements, types of output produced, user-friendliness of the program, temporal and spatial ranges of validity, and accuracy of predictions, are among the many criteria that determine model choice.

Model inputs Model inputs can be classified as data that are likely to be measured for the specific task under investigation, data that can be obtained from previously published information, and data that can only be measured or estimated with some difficulty. The latter can occasionally be estimated by model calibration. If we ignore the validity of using the data to be forecast to derive input parameters to make those forecasts, model calibration can only be used when the end result and all other input parameters are known. This is only possible with research models, which are generally not useful for management or true predictive forecasting. Weather data which cannot be forecast from previous information will always be a problem.

Model outputs Where pesticides are concerned, models can be used to predict transport and persistence of residues in soil, and to predict the potential pollution of surface and ground water. They can also be used as an aid to environmental management at the farm scale, and for risk assessment at the regional scale.

User-model interactions Any program can be easy or difficult to use, can need few or many input data, can require data that are easy or difficult to obtain, can use little or much computer time, can demand input data from the user alone or can search appropriate data bases, can be user-friendly or unfriendly. A brief summary of the ease of use of the three main models used in this research project is given in Table 3.1. This list is somewhat subjective, but attempts to define ease of use in terms of several criteria.

Any user-friendly program must have the following features, for example: a help facility, easy movement and navigation through the various input routines, and data checking for internal consistency. Whilst of lesser significance in scientific environments, user-friendliness may be very important in other situations. It may be argued that models should not be made too simple to use, since only those users who are able to overcome the difficulties set by the programs will have the knowledge to appreciate the intrinsic weaknesses of the models.

The user interface MARVEL, developed as part of this project, can manage a number of models and features on-line help and range-checking of input data. This is the type of interface that should be produced in the future, providing aid to both expert and less-than-expert users. MARVEL is of considerable value since it optimises the time spent on input of modelling data.

The ability to search data bases is another important feature of a user interface, and this is an essential component of the PEMOSYS system developed, in part, at BBA Braunschweig within this project. Data bases can be searched to provide chemical properties, default values (or values from similar sites) for weather and soil data, and to suggest ranges of possible values etc.

The run time of models is also an important aspect of model/user interactions. Long run times may not be restrictive for research purposes, but are not appropriate for management or legislative decision-making. Improvements in structuring and writing of program codes may well accelerate model execution and hence extend their usefulness

Table 3.1 Ease of model use may be assessed by various criteria, as shown in the following Table. The number of + or - signs indicates the classification, with ++ (very good), + (good), - (poor), and -- (very poor).

	Assessment criterion	VARLEACH	LEACHP	PRZM2
1	Number of required input data and their availability by measurement or estimation	+	-	-
2	Ease of data entry/preparation of the input file	++	++	--
3	Ease of run repetition or modification of input files	++	++	++
4	Run time	++	-	++
5	Quality of output			
5a	Comprehensiveness	+	+	++
5b	Clarity	+	+	-
6	Number of phenomena taken into account	+	++	+
7	Quality of the manual/documentation			
7a	Theory and equations	+	++	+
7b	Data values - fixed, default and approximations	+	-	+
7c	Help with estimation of missing input values	-	+	++
7d	Sensitivity to changes in input parameters	-	-	-

Output and data presentation Simple numerical or commented outputs are currently available with most models, but rarely provide graphical display of results. As hardware becomes more standardised, it is hoped that more graphic interfaces will be made available, as is the case with MARVEL. Most simple numeric outputs are produced as a starting point for graphical plotting (as, for example, in VARLEACH); commented outputs are particularly useful for checking data consistency, model operation, and for assessing results and model status during a run.

Documentation Precise documentation is an essential component of user-friendliness, but is often not available or up to date. It can be very difficult, even for an expert user, to follow program structure and to identify critical parameters that are set internally within a program. Often paper documentation is not updated at the same time that the program code is modified, and sometimes it refers to other published papers, thus increasing the difficulty of understanding the program structure.

Real world/model relationships Relationships with the real world can be studied by means of a number of tools, such as sensitivity analyses, error propagation tests (or uncertainty analysis), stochastic validation, and validation with real data sets. Only validation with real data (the type of exercise reported in Section 2.2.5 above) is a true comparison with the real world. Unfortunately, uncertainties and variability within field data sets can be so great that they cannot be used for rigorous model testing purposes, and in reality, they only give a qualitative indication of the accuracy of simulations.

3.1.2 Sensitivity analysis

Sensitivity analysis (see Section 2.2.6) is a technique for testing models - by examining the effects of changes in input parameter values on model output . When parameters are internal to a model, this type of procedure is referred to as "ruggedness" analysis, testing the stability of the model to alterations of its operating procedures.

Error propagation tests (Uncertainty analysis) Modellers are frequently confronted with uncertain input data and it is an important part of model testing to relate the range of model responses to the error range of input data.

Stochastic validation This is a form of sensitivity analysis that involves changes to more than one variable simultaneously. If such combinations of parameters can be selected at random, a statistical analysis of the output is possible.

3.2 SPECIFIC COMMENTS ON INDIVIDUAL MODELS

None of the models tested in this project was perfectly suited to the task of predicting both residues in soil and leachate concentrations. Some drawbacks are due to the way the processes in soil are described, some to the unavailability of input data and some to the extreme sensitivity of the models to certain input parameters. In the following, a more detailed description is given for each of the models tested.

LEACHP

The processes in soil are described in great detail in LEACHP, but this requires many input parameters, several of which are difficult to obtain. This can lead to misleading or inaccurate simulations if parameters are changed without good knowledge of the model setup. For example, the initial soil temperature is used as the lower limit for the soil temperature over the whole simulation. This is a serious limitation, especially for simulations starting in summer.

If the application depth for the pesticide is set to 0, considerable quantities of the chemical (depending on the solubility) may remain on the soil surface and be unavailable for degradation until there has been sufficient volume of rainfall to dissolve the applied dose.

Great care must be taken with input parameters for dispersivity; a default value of layer thickness is a first approximation.

PRZM2

Some processes in soil are not described adequately. Soil temperature has no influence on degradation, but is used to simulate volatilization. Therefore a field degradation rate has to be used as an input parameter. The input parameters for the implemented model for biodegradation were impossible to obtain from the datasets which were collected in the database, and indeed are not available in the literature for other than a few highly specific soil/pesticide combinations.

Soil temperature can be estimated by the program, but the input parameters used (albedo, reflectivity of the soil, windspeed, monthly temperatures at the lower border of the soil core and initial temperature of each soil layer, thermal conductivity, and heat capacity of the soil) are all difficult to obtain. Evaporation can also be estimated by the program using monthly daylight hours and temperature, but this method is too simple and inaccurate. Water extraction by evaporation from soil does not include upward movement in soil but is simply an extraction of equal amounts from each soil layer down to a specified depth. In many instances, PRZM2 predicts greater leaching than either LEACHP or VARLEACH, and this probably reflects this lack of upward transport when evaporative losses are high.

Some parameters had an unexpectedly important influence on the simulation results. The value of the hydrodynamic dispersion coefficient (which is very difficult to estimate) had a tremendous effect on predicted distribution and leaching, and even very small changes in its value produced very different results. It should therefore be used very carefully, if not set to zero.

VARLEACH

This model requires far less input parameters than the other two and is therefore much easier to handle. On the other hand, this leads to some limitations. Application depth is always assumed to be 1 cm, and plant growth is not included. There is no limitation for water entering the soil; in the case of very high rainfall events or high irrigation, this leads to very deep leaching. Calculations are always made in 1 cm increments, and other depth increments are used only for the output. This has the advantage that there is no influence on total residues, in contrast to the other models. Up to 3 soil layers, with different properties, can be used for the simulation. The depth where changes occur in adsorption, degradation, bulk density and field capacity has to be the same for all these input values.

Pan evaporation is an essential requirement for valid results with VARLEACH, because the evaporation reduction routine implemented is based on measured values. It was found that even after calibration, the Linacre estimation did not give sufficiently accurate daily estimates, especially in winter when the calculated values were much larger than those measured.

LEACHP, PRZM2 and VARLEACH

The sensitivity analysis revealed that degradation and sorption parameters of the chemical had the strongest effect on the simulation results. It is therefore important to determine these values carefully, and if possible under conditions comparable to those which occurred in the experiment to be simulated. The sensitivity of all models to these parameters was very similar.

Most soil parameters did not have a great influence on the results, but there were exceptions with some models. In LEACHP and PRZM2, sorption is calculated from the organic carbon content of the soil, which is therefore very important. It should be noted that some of the soil parameters which are difficult to obtain (dispersivity, hydrodynamic dispersion) or model parameters (depth increment) can have an enormous effect on the results. Therefore, if these parameters are not known, it is important to use the default values stated in the manual.

Overall, the performance of the models in comparison with field data sets was not satisfactory. Although in some cases explanations for the poor fits can be given, the use of models as predictive tools is still very unreliable. However, since they all have very similar sensitivity to most input parameters, they may be used for relative assessment of compounds in comparison with others under standard conditions.

3.3 CONCLUSIONS FROM THE MODEL EVALUATION EXERCISE

When the selected data sets (10 from Germany, 6 from the UK, 6 from France and 4 from Italy) were examined in detail using the models VARLEACH, PRZM2 and LEACHP, all of the models gave excellent correspondence with observed data in some circumstances (which cannot be clearly defined), and very poor fits in others. Even within time-phase samples from a single experiment, there was considerable variability in model performance. One main conclusion was the general lack of appropriate data for strict model evaluation since much of the essential information was not available. Generally there was also poor description of the variability in the observed data, thus making true evaluations of goodness of fit impossible. The model output data were always compared graphically with those observed which gave a subjective indication of the goodness of fit. In several instances, observed and predicted data were compared using the statistical indices described above. Based on these statistical criteria, LEACHP generally gave the best overall prediction of leaching, whereas VARLEACH appeared to have the best subroutines for prediction of degradation. However, differences between the models in terms of the statistical comparisons were often small. All three models gave reasonable assessments of soil residues, but predictions of leachate concentrations were generally poor, particularly with respect to the timing of breakthrough. It therefore appears that the models are adequate to describe the processes for which they were developed initially - the redistribution and persistence of residues in soil. They are of less use for predicting likely concentrations in drainage water, other than in terms of the relative pollution potential of different compounds (see Section 2.2.7). Clearly, better process description of factors such as preferential flow is essential, and much current modelling effort is devoted to simulation of this important process.

3.4 ACHIEVEMENTS FROM THE RESEARCH PROJECT

The main achievements from the research project can be summarised:

1. A fully-indexed data base of results from pesticide leaching studies has been developed. This is freely available to any group that requires data for model evaluation or any other purpose.

2. A user-friendly interface for input of data into the models VARLEACH, LEACHP and PRZM2 has been developed. This is already in use by members of the research community in the UK and by the Registration Branch of the UK Ministry of Agriculture Pesticide Safety Directorate. A list of recipients for wider distribution has been collated.

3. A Software package (SIMUL) has been developed to monitor the flow of information through the VARLEACH, LEACHP and PRZM2 models. This defines the programs in modular units and indicates ways in which the models can be recombined to allow sub-routines from different models to be used.

4. A new program (ENGINE) has been developed to run the modular units in logical paths chosen by the user. Modular units can be assembled that derive from different models. A particular benefit is that the system, when perfected, should permit new subroutines to be easily incorporated into existing models.

5. A FORTRAN program with the name STATIND has been written to calculate a number of statistical indices as an aid to the interpretation of the comparison between observed and predicted data.

6. The part of the research programme involving detailed modelling exercises has resulted in eight contributions to Scientific Conferences and Symposia and to three Refereed Journal papers (See Section 1). Other publications are in preparation.

3.5 FUTURE NEEDS

Software development

Two software packages called MARVEL and ENGINE have been developed for this project. MARVEL facilitates data entry, running and presentation of results in three widely-used models. ENGINE divides the same three models into a modular structure with the objective that the best components for a given scenario from any of the three can be selected and combined to give improved simulations. ENGINE is relatively complex and it would be desirable to link this software with MARVEL to facilitate its wider application.

Experimental data

One of the main problems encountered in this project was the absence of high quality data to use for model evaluation. Most of the experimental results reported in the data base were not collected for model evaluation purposes, and hence they were often incomplete in terms of the parameters required as inputs for the different models. An additional deficiency was the general lack of information concerning variability in both the input parameters and the measured residue data. It is important to maintain the data base and to add the results of further, more complete studies as they become available. An alternative possibility for the future would be to establish a European network of well-equipped field sites covering a range of climatic, hydrological, pedological and cultural conditions. All appropriate soil, hydrological, and pesticide data could be obtained for the different sites, with appropriate attention paid to spatial and temporal variability.

Modelling

There is a need for further evaluation and refinement of accurate process-based models. Guyot (1994) suggested that there is no pressing need for development of new models, but there is a need for more detailed evaluation and validation of the various models that are currently available. Further refinement in certain process descriptions is also required. Limitations to current models include poor descriptions of volatilisation, the general absence of routines to predict preferential flow of water and solute, the poor (and often absent) description of time dependent sorption processes, and the simplified ways in which the kinetics of degradation are described.

This project has focused on leaching of pesticides through soil as one of the main routes for contamination of water systems. Direct runoff into adjacent ditches and surface water courses is a second route for contamination of water systems. Modelling of surface runoff and associated chemical transport has received less attention than leaching through soil, even though the resultant concentrations in water may be significantly larger. Future work should address the state of the art of runoff modelling through evaluation against new or existing data, and should make recommendations for aspects requiring further development.

Vulnerability assessment

Although accurate process-based models are a pre-requisite for prediction of site-specific patterns of behaviour, and for gaining a better understanding of the interactions and importance of different processes in different circumstances, there is also a need to extend their use to larger regional scales. One way to account for the wide variability at regional scales is to use input in the form of probability distributions and to produce a stochastic model output. This once more raises the need for information on the variability to be expected in input parameters, and on how to deal with co-dependent parameters.

Simple assessments

In addition to the detailed regional assessment of vulnerability to pesticide contamination, there is also a need for simplified procedures that take account of climatic, hydrological, soil and cultural characteristics for ranking pesticides with regard to their potential to pollute surface and ground waters in different situations. Some simple systems are already available but as with the detailed models, they require further refinement to extend their use. The simple "Groundwater Ubiquity Score" or "GUS index" suggested by Gustafson (1989), for example, can take account of degradation and mobility characteristics of the pesticide, but would be more useful if it could be extended to take some account of soil and hydrological characteristics.

References

Gustafson D I (1989) Groundwater Ubiquity Score: A simple method for assessing pesticide leachability. *Environmental Toxicology and Chemistry,* **8**, 339-357.

Guyot G (1994) Strategies to minimise the pollution of water by pesticides. Pages 87-152 in: Pesticides in Ground and Surface Water, H Borner (Ed). Springer Verlag, Berlin.

APPENDIX I: Data Base Entry GB01

EXPERIMENT TITLE: "Persistence and mobility of Alachlor in soil in UK"

SUMMARY: A suspension of Alachlor was applied to the surface of 7 replicate soil columns in a mini lysimeter system. One column was removed immediately after application of the herbicide and single columns were removed at intervals of approximately 28 days during the subsequent 168 days.

Alachlor concentrations in successive 2cm segments of each column were measured. Concentrations in the leachate were also measured after significant rainfall.

DURATION: 5 November 1990 - 22 April 1991

TYPE OF EXPERIMENT: lysimeter with undisturbed soil
11 cm diameter, 30 cm depth

SITE: Hunts Mill, HRI Wellesbourne, Warwickshire

altitude: 48 m above sea level latitude: 52.205 degrees N

no groundwater

no crop

no slope (flat land)

IRRIGATION: no irrigation

EXPERIMENT: applied on 5 November 1990

8 kg/ha, commercial emulsifiable Alachlor (48%),
together with analytical grade Alachlor
(British Greyhound Ltd, Birkenhead)

surface application

sampling date: 5/11/90 3/12/90 2/01/91
28/01/91 25/02/91 25/03/91
22/04/91

WEATHER: maximum/minimum temperature (°C)
(daily)
rainfall (mm)

pan evaporation (E0), (mm)

(recorded from 1st September 1990 to 30th April 1991 in file GB01_03.MET)

SOIL: Eutric Cambisol

soil characteristics for upper 0 - 30 cm:

water content at 5 kPa: 13.41 (ml water per 100 g dry soil)

water content at 1500 kPa: 4.7 (ml water per 100 g dry soil)

particle size distribution: Sand 76%, Silt 10%, Clay 14%

soil texture: sandy loam

organic carbon content 1.12 %

bulk density (kg/l) 1.38

saturated conductivity (cm/day) non available

microbial biomass: 206.9 mgC/kg

microbial respiration: 5.63mgC/kg/day

CHEMICAL: water solubility: 242 mg/l (at 25 °C)

vapour pressure: 187·10-5 Pa
(from Reviews of Environmental Contamination
and Toxicology, Volume 123, Springer-Verlag)

Henry constant: 1.3 10-6 (dimensionless)
(from RUSTIC User's Guide Volume II,
Environmental Protection Agency,
Athens GA 30613 USA)

Kd in the soil used in the experiment: 1.07 (l/kg)

adsorption isotherms (Kfr (l/kg), 1/n): non available

LABORATORY RESULTS:

Temperature (°C)	Half life (days) at 12% soil moisture	Soil moisture (% w/w)	Half life (days) at 15°C
5	119	12.02	46
10	77	7.34	62
15	40	5.52	81
20	15.7	4.25	200
25	17.3	3.82	238

FIELD RESULTS

depth (cm)	soil residues (μg/kg of dry soil)						
	0	28	58	84	112	140	168 (days)
0 - 2	28985	12030	2452	1759	1997	1994	1145
2 - 4	0	7090	3264	2075	2455	1959	953
4 - 6	0	1322	4490	3188	3478	1980	1409
6 - 8	0	0	3284	3678	2612	2101	1681
8 - 10	0	0	1275	2890	1403	1307	1104
10 - 12	0	0	664	1209	1113	614	635
12 - 14	0	0	368	852	887	325	264
14 - 16	0	0	261	461	559	310	130
16 - 18	0	0	200	391	293	261	55
18 - 20	0	0	145	183	78	-	-
20 - 22	0	0	96	43	-	-	-
22 - 24	0	0	55	0	-	-	-
24 - 26	0	0	58	0	-	-	-
>26	0	0	26	0	-	-	-

Days	Average volume (ml)	Leachate Concentration (μg/litre)
16	68	0
39	283	51.1
53	153	2.3
58	60	0
66	284	3.2
68	18	4.7
78	49	25.2
113	126	17.1
129	187	4.1
141	54	0
155	57	0

Appendix II - The Engine structure

AII.1 The programming language

C++ has been chosen as the programming language.

The language C++ is an extension of the C language which has become a standard. Selection of C++ language is the result of the following considerations:

- it is extremely fast
- it is the most frequently used language in software development
- data encapsulation and protection are allowed
- selection and pooling of variables are enabled
- arrays can be encapsulated in an orderly interface
- operator overloading enhances readibility of mathematical formulae involving arrays and matrices.
- "function pointer" variables are available

A number of applications of these points are discussed below.

Object-oriented development - the future of C++ software ?

C++ contains object-oriented features, such as inheritance, data protection, polymorphism and operator overloading. However, this programming style has not been fully exploited, as a functional breakdown of models was preferred - as discussed above with reference to the Simul software and variable flow-diagrams. Nevertheless, object-oriented features have been used when they clarify module- or engine interface- code.

AII.2 Structure of the program

The Engine ("main")

The principal objective was to create an environment in which researchers can assemble simulation programs in modular fashion.

A number of developmental stages have already been discussed:

1) Splitting programs into modules, each one of which emulates a well-defined fundamental physical process (using the Simul program).
2) Clarifying the interactions between modules (using SimulGraph).
3) Identifying and collecting (into a separate program) section modules which act only on program flow.

Therefore, the new environment is comprised of the following components:

- a *List of Routines* implementing micromodules
- a *Simulation Engine* (the job scheduler)
- a *Set of variable pools* shared by the Engine and the micromodules

AII.2.1 The routine list

The list of routines is stored in a file that the user can read and write using any word processor. It is easily upgradeable and contains the list of routines to be used in the simulation. In the future, the list will be compiled by the user through a user-friendly interface.

AII.2.2 The job scheduler

Since each simulation step is time-based, a time-managment routine is required. The new Engine must implement the time-loop style of event driving, with more flexible time managment. The main program of the Engine is essentially a job-scheduler for the micromodules.

Time managment: the clock

Event-driving is coordinated by an internal simulation clock. The user can specify the duration of the time step and the timing of execution of each micromodule in terms of the clock steps. The internal clock is accessed by the micromodules to determine the current time, the execution status of all modules, and eventually to actively alter their scheduling times.

The micromodule scheduling table

The user can modify the scheduling of each micromodule by specifiyng :

1) The initial time of execution
2) The frequency of execution
3) The order of execution

All this information is stored in the executable table.

An example from the VARLEACH implementation of the Engine is given below:

Micromodule	**Starting Date of execution**	**Frequency of execution**
Rain	Day 1	Daily
Evaporation	Day 1	Daily
WaterMovement	Day 1	Daily
ChemicalTransport	Day 1	Daily
Degradation	Day 1	1/20 day
DailyReport	Day 1	Daily

Modules dedicated to flow control

An important part of the software implementation of simulation models is the control of program flow. This point is clarified by examining models that solve differential equations, such as LeachP or PRZM2. For instance, the subroutine *TStep* of LeachP is dedicated to calculating the time step for the next solution of the water movement equation (performed by Watflo).

We define control modules of the Engine as the modules which interact with the micromodule scheduling table in order to read or change the execution times and frequencies.

Calls to the Engine interface

Some special calls have been programmed, allowing exchange of information between the engine and the clock. As in other program sections, these have been given relevant names which are readily understood. Some of these calls are listed below.

```
Class: Engine
void   SetSimulationDuration(long  d);
void   SetSimulationStartTime(long  startTime);
long   GetModuleNextExecutionTime(char  *moduleName);
long   GetModuleExecutionPeriod(char  *moduleName);
void   SetModuleNextExecution(long  newNextExecution);
void   SetModulePeriod(long  newPeriod);
```

```
Class: Clock
long   DaysElapsed();
long   UnitsElapsed();
long   UnitsInDay();
float  DayFractionElapsed ();
```

AII.2.3 The variable Pools

Managment of global variables by the Pools

The Engine and the micromodules can be separated if the Engine does need not to know names or values of variables used inside micromodules. In a standard structure that cannot be achieved, because:

- when variables used in modules are **local**, then other modules cannot define them
- when variables are **global**, they need to be declared in the main, ie. made made known to the Engine.

One way of overriding the problem is to define global objects in the "main" program (Engine) that are not variables but lists of variables, or **Variable pools**. These have the following features:

- Variables can be of any standard type (float, int, string etc...)

- The Engine defines which variables will be contained within the lists at runtime, dependent upon which micromodules are loaded (i.e. listed in the routine list)

Each Variable Pool consists of a list of similar variables indexed by name, and also the pool-class methods. When adding a micromodule to the program, the pool content is updated, if necessary, with the names of the new global variables. Variable values can be accessed by means of calls made to the methods of Pool class (see below). The mechanism of creating variable pools from global variables will be clarified below, in the examples given in the discussion of micromodule libraries.

Initially, one large pool was created, containing all the variables shared by the micromodules. However, it became clear that it was better to define a number of pools, with one listing chemical properties, another listing soil or weather properties etc. In this implementation of the Engine we have the following pools:

- simulation
- chemical
- soil
- weather
- scenario

Variable Pool data types

All standard numeric types of C language are permitted in this implementation, ie. Integer, Float, Double.

In order to deal with soil layering, the following array types were defined:

- Farray (array of float numbers, dynamic)
- Darray (array of double numbers, dynamic)

Names of input files and other alphabetical requirements are dealt with by means of the array:

- String (array of characters, dynamic)

All arrays are dynamic (i.e. size can change at run-time). Standard C notation is used to access their values. For example, when access is required to a variable named waterContent in the scenario pool, the statement

```
waterContent[3] = waterContent[3] + 10;
```

increments the water content of the third layer by 10 units.

Pool-class calls

The Pool class has defined methods of accessing variable values. Some of these calls are listed below:

```
Class: Pool
float       GetFloatValue( char* );
int         GetIntegerValue( char* );
FArray      GetFArrayValue( char *str );
String      GetStringValue( char *str );

void        SetFloatValue( char*, float );
void        SetIntegerValue( char *, int );
void        SetFArrayValue( char*, FArray );
void        SetStringValue( char *name, char* val );
```

AII.3 micromodule libraries

Micromodule structure

Every micromodule contains lines of program which perform the physical simulation implemented. In order to access values of global variables, the micromodule calls the appropriate pool with a pool class method. For example, when micromodule calculations start from the half-life of the chemical in the soil, the micromodule questions the chemical pool by writing:

```
float hl;
...
...
hl = chemical.GetFloatValue(  "halfLife");
...
...
```

where:

float hl = a definition of a **local** variable (hl) in which a copy of the value stored in the halfLife **global** variable is made;

... = a definition or calculation

```
hl = chemical.GetFloatValue(  "halfLife" );
```

is the actual call by which the copy in the **local** variable is made.

The half-life value can be used by the micromodule by means of the **local** variable hl. If the module changes the half-life value in hl, then the value of the global variable halfLife must also be changed, hence:

```
float hl;
...
hl = chemical.GetFloatValue(  "halfLife" );
...
hl = hl ^ 2 - 3;
...
chemical.SetFloatValue(  "halfLife", hl );
```

where:

```
hl = hl ^ 2 - 3;
```

is a summary calculation, and

```
chemical.SetFloatValue(  "halfLife", hl );
```

is the call by which the half-life value stored in the pool (global variable halfLife) is updated. Subsequent calls to GetFloatValue("halfLife") will return the new value.

This method may seem somewhat overcomplicated, but does provide a number of advantages:

- It is perfectly clear which micromodules are required and which global values are used
- There are no restrictions to the names which the micromodule writer gives to variables specific to any micromodule, as no interference occurs with global names
- Values of global variables can change only by means of a function call. This permits implementation of **micromodule libraries** and **separate compilation of libraries**. This technical advantage will be discussed subsequently in the section: "micromodules and the windows DLL libraries".

An example of micromodule implementation is given below.

The module is the temperature calculation of a soil compartment, as performed by VARLEACH. Lines beginning with // are comment lines.

```
void  Temperature()
{

   //
   // import time status
   //
   int   cDay = engine.clock.DaysElapsed();
   float dfrac = engine.clock.DayFractionElapsed();

   //
   // import vars from pools
   //
   FArray   maxT = weather.GetFArrayValue(  "maxTemperatures"  );
   FArray   minT = weather.GetFArrayValue(  "minTemperatures"  );
   int   lNum = simulation.GetIntegerValue(  "layerNumber"  );
```

```
    //
    // local vars
    //
    float    airMeanTemp, airTempAmplitude;
    float    soilMeanTemp, soilTempAmplitude;
    float    a3;
    FArray   temp( lNum );

    airMeanTemp = ( maxT[ cDay ] + minT[ cDay ] ) / 2.0;
    airTempAmplitude = ( maxT[ cDay ] - minT[ cDay ] ) / 2.0;
    soilMeanTemp = 1.7 + 0.994 * airMeanTemp + 0.466 * airTempAmplitude;
    if( airMeanTemp < 7.0 )
       soilMeanTemp = 0.817 * airMeanTemp - 0.28;
    soilTempAmplitude = 0.53 + 1.62 * airTempAmplitude;

    // calc temperature for all layers
    for( int i = 1; i < lNum - 1; i++ )
    {
       a3 = soilTempAmplitude * exp( - ( i - 0.5 ) / 9.3 );
       temp[i] = soilMeanTemp + a3 * sin( 3.142 * ( 1.5 + 2 * dfrac ) );
    }

    //
    // output section
    //
    scenario.SetFArrayValue( "dayFractionTemperature", temp );

}
```

Some code-line groups can be identified:

- A section (labelled "import time status") in which the routine calls the clock to determine the current simulation time. The clock gives the current day of simulation and the fraction of current day elapsed (floating point variable).
- A section ("import vars from pools") in which necessary values are copied from variable pools.
- A section ("local vars") in which local variables are defined and some preliminary calculations are made.
- A section ("calc temperature for all layers") in which values are fitted in the local variable 'temp', which is an array equal to layer number.
- An "output section" in which only the temp array is used to change the array global variable "dayFractionTemperature" in the scenario pool.

This method of communication between micromodules and the Engine represents one possible scenario. Another possibility would be to use a call in the input section, directed to the scenario pool to get a 'handle' (pointer) on the value of the array "dayFractionTemperature", thereby avoiding copying the array 'temp' into the global array "dayFractionTemperature". This would be more rapid, but at the cost of reducing the clarity of the micromodule/engine interface. The code could therefore have been written as thus:

```
// import section
...
Farray temp = scenario.Use( "dayFractionTemperature");
...
// calculations
....
```

The values of the array "dayFractionTemperature" would have changed without requiring an export call. Use of the array 'temp' as an alternative would be very powerful in terms of speed, but those looking at the micromodule would not know whether the temperature has been **used**, **assigned** or **both**, without reading details of the calculation section.

Separate compilation of micromodules by means of the Windows Dynamic Link Libraries
The micromodules are entities separated from the "main" part. This led to consideration of the possibility of a separate compilation of these modules. This has been tested in the Windows 3.1 environment by means of the Windows Dynamic Link Library (DLL). Separate compilation could enable developers of new micromodules to be consider their routine implementations alone. However, the DLL feature will be present only in future releases of the Engine software.

The Windows environment has been selected for future development of the Engine for the following reasons:

- Requirement of a graphical interface for the Engine in future developments
- The DataBase is presently running under Microsoft Access.

AII.4 Implementation of existing models in the Engine structure

AII.4.1 The VARLEACH program

VARLEACH loop structure

The structure of time and layer loops in the VARLEACH program is implemented as follows:

```
...
...
day loop
{
    Rain
    Evaporation

    boundaries loop
    {
        for the upper compartment:
        Water Content
        ChemicalConcentrationDistribution
        day fraction loop
        {
            Temperature
            Chemical Degradation
        }
        for the lower compartment:
        Water Content
        ChemicalConcentrationDistribution
    }
}
```

In our implementation, the boundary loop was inserted within the processes of water and solute transport. In this way, there is no need for a layer loop, and water and solute tranport processes can be connected to modules extracted from other programs.

Models implementing Richards equation or the CDE give other problems. These are differential equations, the solutions of which are space arrays similar in type to waterContent[.] or chemicalContent[.]. It is acceptable to have these arrays as the input/output value storage for the micromodules. This can also be done for VARLEACH, provided that the water transport module has the flux array and the old-water array as output values. In this way the chemical transport micromodule can calculate the correct values for the concentrations while water is moving down the layers.

Running VARLEACH through the Engine

This is the content of the input file required for running VARLEACH through the Engine program:

```
#
# Module Execution Table
#
# Name                  Frequency    Start
ReadDataSet              0            0
ArraysInit               0            0
ComputeDegrConst         0            0
TopLayerChemical         0            0
TopLayerMoisture         0            0
WeatherDataCorrection    0            0
BDCorrection             0            0
DailyOutput             20           20
AddRain                 20            0
Evaporation             20            0
KdTimeVariation         20            0
ChemicalPartition       20            0
WalkerWaterEngine       20            0
ChemicalTransport       20            0
Temperature              1            0
Walker                   1            0
```

"Timing" of evaporation and degradation routines

In VARLEACH, evaporation, temperature and degradation routines are run many times per day. This is implicit in the structure of the model for temperature and (Walker) degradation routines, with the latter using temperature values which change every 1/20th of a day. The evaporation routine can be internally divided into 1/10th fractions of a day, but there is no need to assign it a frequency greater than 1 day. This is because no other micromodule interacts with water content during evaporation. If the evaporation period was changed to fractions of a day, then a different model would be obtained; the chemicals would degrade in the first layers during the individual evaporation steps and a different rate of degradation would be calculated.

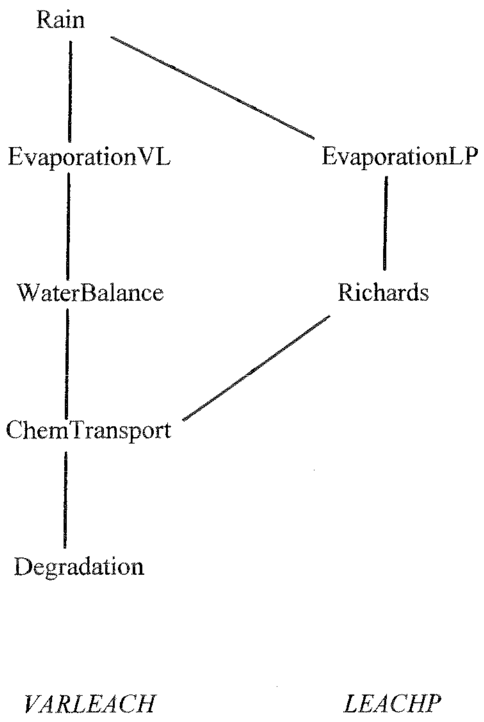
AII.4.2 The LEACHP program

Implementation of Richards equation

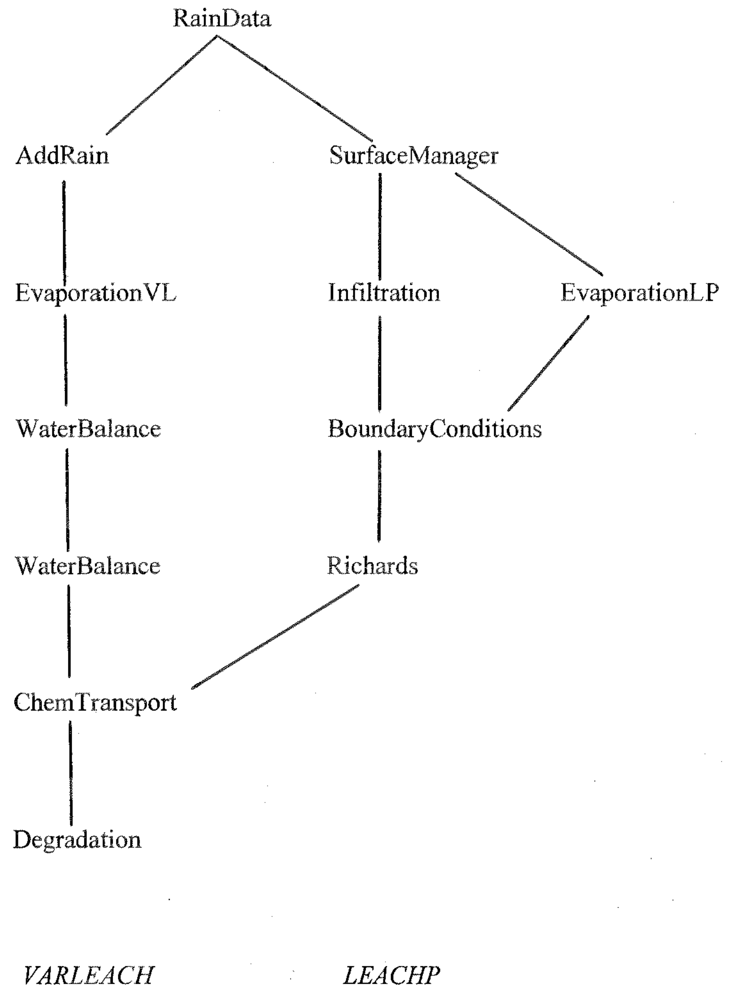
In the implementation of LEACHP there is a new 'timing' structure. The Richards equation solution (provided by the Watflo routine in LEACHP) uses an adaptive time-step procedure during solution of the differential equation. This procedure is capable of dealing with time steps of very small fractions of a day. This has led to development of a "clock" with user-modifiable time-units. The clock is responsible for indicating the number of elapsed days at any time. Variable time steps have been incorporated by means of a control-module, similar to the routine Tstep in LEACHP. This routine controls the duration of the time step by analyzing both the timing of irrigation events, and flux densities computed during previous steps of calculation.

Interconnections with VARLEACH modules

Since both VARLEACH and LEACHP have a WaterTransport and a Solute Movement routine, it may be possible to connect these two models:



This is the first attempt to connect these models. Translation of water content to water potential is required, as Watflo requires input in the latter format. However, a problem arises with LEACHP, where the reference manual warns of instability of solutions when water fluxes increase. This means that linking the equation solving routine to a routine that simply adds rain to the first soil layer would lead to incorrect results. In addition, the LEACHP extension of the Campbell equation, used to translate water contents to potentials (function Potl), is singular when water content is greater than field capacity. So, it is not possible to add rain or to permit evaporation without controlling the fluxes (positive or negative) across soil levels. In LEACHP, this is achieved by means of form of "surface manager" (variable IHEAD) that switches between infiltration, ponded water or evaporation, depending on rain events. This situation can be illustrated:



On account of the prolonged duration of the debugging process for the restructured version of VARLEACH, it was not possible to translate this routine into our environment during this current project. However, work is currently underway to include this implementation, which will enable full advantage to be taken of the two different "water engines".

SUMMARY

Evaluation and improvement of mathematical models of pesticide mobility in soils and assessment of their potential to predict contamination of water systems.

In this project with participants from Germany, the United Kingdom, Italy and France, the simulation models LEACHP, PRZM2 and VARLEACH were examined with regard to their efficiency in predicting the contamination of water (including groundwater). For this purpose, about 80 data sets from the participating countries were stored in a database, which also provides most of the input data necessary for modelling, e.g. soil and climatic data. This comprehensive pool of data from contrasting climates offers the possibility to investigate the suitability of the models under varied conditions.

A user-friendly interface (MARVEL - Model Attribute Relationship Validation and Entry Layout) was created to facilitate use of the three main models LEACHP, PRZM2 and VARLEACH. It allows data input a sequence of standardized screens, from which model input files (e.g. input or weather data files) are created automatically. Additionally a routine for graphical output of different results is included.

Measured and simulated results (from LEACHP, PRZM2 and VARLEACH) using 26 selected data sets with comprehensive input data were compared with respect to distribution of residues in soil and, where possible, leachate concentrations. No general conclusions can be drawn regarding the suitability of the models. Even within time-phase samples from a single experiment, there was considerable variability in model performance, with no obvious explanation. Using several models on one data set, each might give the best fit at a different sampling date. Overall, redistribution and persistence in soil were described better than breakthrough and concentration in leachate.

As an aid to the model evaluation exercise, a number of statistical indices were identified and brought together in a programme (STATIND) for data comparison. Model efficiency, for example can be used to evaluate the overall goodness of fit, whereas other indices can be used to compare leaching depth or total residues in the soil profile.

In a sensitivity analysis of the models LEACHP, PRZM2 and VARLEACH, the influence of different input parameters on the simulation results was investigated. In all three models, pesticide

degradation and sorption parameters had the greatest effect on model results. In addition, some parameters which are difficult to obtain or not directly related to data sets, such as dispersivity, initial soil temperature, or layer thickness, can have major effects and values for them should be chosen with care.

This was also shown in a ring test with LEACHP, PRZM2 and VARLEACH, where all five groups participating in the project were given full information from a comprehensive field experiment, but without the measured results (concentration in soil and leachate). No two sets of predicted results for a given model were exactly the same. The few unknown input data (initial temperature, layer thickness) caused a variation in output of a similar order of magnitude as in the observed results, but smaller than the discrepancies between observed and predicted data.

For further analysis of the models, programmes were developed to identify variables and routines in the models and their sequence of use and operation during the course of simulation (SINTAX, SIMUL). Based on this work is the programme ENGINE, which contains routines (e.g. for degradation, sorption, water transport) from the models LEACHP and VARLEACH as independent blocks. PRZM2 is to be included in the near future. Now routines from different models can be combined for one simulation, e.g. degradation from VARLEACH and water transport from LEACHP.

The results of this project financed by the European Commission can be used to further optimise the use of simulation models for the assessment of the behaviour of plant protection products within the EU. In contrast to the situation in Germany, where only PELMO is recommended as an assessment model, the „Uniform Principles for Assessment and Registration of Plant Protection Products“ propose the use of any model which is acknowledged by the EU.

ZUSAMMENFASSUNG

Überprüfung und Verbesserung von mathematischen Modellen zum Einwaschungsverhalten von Pflanzenschutzmitteln und Beurteilung ihrer Anwendbarkeit zur Vorhersage der Kontamination von Gewässern

In diesem Projekt mit Teilnehmern aus Deutschland, Großbritannien, Italien und Frankreich wurden die Simulationsmodelle LEACHP, PRZM2 und VARLEACH auf ihre Eignung zur Vorhersage möglicher Risiken einer Gewässerkontamination (einschließlich Grundwasser) untersucht. Dazu wurden ca. 80 Datensätze für Feld- und Lysimeterversuche aus den Teilnehmerländern in einer Datenbank zusammengefaßt, in der auch die meisten für die Modellierung notwendigen Eingabedaten (z.B. Klima- und Bodendaten) zur Verfügung gestellt werden. Dieser große Datenbestand aus sehr unterschiedlichen Klimaten bietet eine Möglichkeit, die Eignung der Modelle für verschiedene Bedingungen zu testen.

Zu diesem Zweck wurde eine Ein- und Ausgabe-Oberfläche (MARVEL - Model Attribute Relationship Validation and Entry Layout) erstellt, mit deren Hilfe die Verwendung der drei wichtigsten Modelle LEACHP, PRZM2 und VARLEACH wesentlich erleichtert wurde. Mit diesem Programm können die benötigten Daten in einer weitgehend einheitlichen Benutzeroberfläche eingegeben werden, und Modelldateien (z.B. Eingabe- und Wetterdateien) werden automatisch erstellt. Eine graphische Ausgabe verschiedener Ergebnisse ist ebenfalls möglich.

Anhand von 26 ausgewählten Datensätzen mit besonders umfangreichen Eingabedaten wurden gemessene und mit LEACHP, PRZM2 und VARLEACH simulierte Rückstände im Boden und z.T. im Sickerwasser verglichen. Dabei zeigte sich, daß keine generelle Aussage über die Eignung der Modelle getroffen werden kann. Selbst bei der Simulation eines Datensatzes mit einem Modell variiert die Güte der Anpassung zwischen verschiedenen Probenahmeterminen stark, allerdings ohne offensichtlichen Grund. Bei der Auswertung mit mehreren Modellen kann jedes zu einem anderen Termin das beste Ergebnis liefern. Generell wurden die Menge und Verteilung der Rückstände im Boden besser vorhergesagt als der Zeitpunkt des Auftretens und die Konzentration der Wirkstoffe im Sickerwasser.

Um die Übereinstimmung zwischen Meßwerten und Prognose zu überprüfen, wurden einige statistische Ansätze untersucht und in einem Auswertungsprogramm (STATIND) zum Vergleich zwischen gemessenen und simulierten Werten zusammengefaßt. So kann z.B. die Modelleffizienz zur Bewertung

der allgemeinen Übereinstimmung zweier Verteilungskurven verwendet werden. Weitere Indices dienen dem Vergleich der Einwaschungstiefe oder der Gesamtrückstände im Bodenprofil.

In einer Sensitivitätsanalyse der Modelle LEACHP, PRZM2 und VARLEACH wurde der Einfluß der verschiedenen Eingabedaten auf die Simulationsergebnisse untersucht. In allen drei Modellen hatten die Abbau- und Sorptionsparameter der Wirkstoffe den größten Effekt. Darüber hinaus können einige Eingaben, die schwer zu ermitteln sind oder nicht direkt im Zusammenhang mit der Datenerhebung stehen (Dispersivität, Bodentemperatur am Start der Simulation, Schichtdicke), das Ergebnis sehr stark beeinflussen und sind daher kritisch zu überprüfen.

Dies wurde auch in einem Ringtest für die Modelle LEACHP, PRZM2 und VARLEACH deutlich, in dem allen Prokjektteilnehmern ein Datensatz mit umfangreichen Eingabedaten zur Verfügung gestellt wurde, wobei die Meßergebnisse (Rückstände in Boden und Sickerwasser) nur dem Initiator bekannt waren. Mit keinem Modell erzielten alle Teilnehmer das gleiche Ergebnis. Die wenigen nicht vorgegebenen Daten (Schichtdicke, Anfangstemperatur des Bodens) genügten bereits, um eine Variation der Ergebnisse zu verursachen, die allerdings in der gleichen Größenordnung wie diejenige innerhalb der Meßdaten lag und geringer war als die Unterschiede zwischen allen gemessenen und allen simulierten Werten.

Zur weiteren Analyse der Modelle wurden Programme entwickelt, die z.B. die im Modell verwendeten Variablen und Routinen identifizieren und die Abfolge ihrer Benutzung im Verlauf der Berechnungen aufzeigen (SINTAX, SIMUL). Auf diese Arbeit baut das Programm ENGINE auf, in dem die einzelnen Programmroutinen (z.B. für Abbau, Wassertransport, Sorption) der Modelle LEACHP und VARLEACH (der Einbau von PRZM2 ist in Arbeit) zusammengefaßt sind. Es enthält für jeden physikalischen Prozeß aus jedem Modell eine getrennte Routine, so daß jetzt z.B. die Abbauroutine aus VARLEACH mit der Wassertransportroutine aus LEACHP kombiniert werden kann.

Die Ergebnisse dieses von der Europäischen Kommission finanzierten Projekts sollen dazu dienen, die Verwendung von Simulationsmodellen zur Bewertung des Verhaltens von Pflanzenschutzmitteln in der EU weiter zu optimieren. Im Gegensatz zur einheitlichen Verwendung von PELMO in Deutschland wird in den „Einheitlichen Grundsätzen für die Bewertung und Zulassung von Pflanzenschutzmitteln in der EU“ kein gemeinsames Bewertungsmodell, sondern die Verwendung eines der auf Gemeinschaftsebene anerkannten Modelle vorgeschlagen.