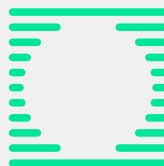


# Benutzerhandbuch zum Programm

## IBJmepod

### Inhaltsverzeichnis

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Übersicht</b>                              | <b>1</b>  |
| <b>2</b> | <b>Installation</b>                           | <b>2</b>  |
| <b>3</b> | <b>Die Benutzeroberfläche <i>IBJmepod</i></b> | <b>3</b>  |
| 3.1      | Erstellung des Eingabe-Files . . . . .        | 3         |
| 3.2      | Rechnung durchführen . . . . .                | 4         |
| 3.3      | Ergebnisse darstellen . . . . .               | 4         |
| <b>4</b> | <b>Das Programm <i>IBJXodor</i></b>           | <b>6</b>  |
| 4.1      | Übersicht . . . . .                           | 6         |
| 4.2      | Eingabe-Parameter . . . . .                   | 8         |
| 4.3      | Beispiele . . . . .                           | 17        |
| 4.3.1    | Fahnenbegehung . . . . .                      | 17        |
| 4.3.2    | Wetterstatistik . . . . .                     | 18        |
| <b>A</b> | <b>Dateistruktur</b>                          | <b>21</b> |
| <b>B</b> | <b>Implementierte HPGL-Befehle</b>            | <b>24</b> |



Ingenieurbüro Janicke  
Gesellschaft für Umweltphysik  
D-26427 Dunum  
Alter Postweg 21

Tel.: (04947) 9120 35  
Fax: (04947) 9120 34  
Email: [info@janicke.de](mailto:info@janicke.de)  
Internet: [www.janicke.de](http://www.janicke.de)



# 1 Übersicht

Das Computer-Programm *IBJmepod* ist eine Realisierung des Geruchsausbreitungsmodells MEPOD. Die verwendeten numerischen Verfahren sind im Handbuch zu MEPOD im Abschnitt *Implementierung* beschrieben. *IBJmepod* stellt eine grafische Benutzeroberfläche dar, die für die eigentliche Ausbreitungsrechnung auf das Programm *IBJXodor* zurückgreift. *IBJXodor* kann auch unabhängig von der grafischen Benutzeroberfläche eingesetzt werden.

Dieses Dokument beschreibt die Installation des Programmsystems, die Handhabung der grafischen Benutzeroberfläche *IBJmepod* und die Eingabe- und Ausgabe-Dateien von *IBJXodor*. Dabei bestehen folgende Konventionen in der Verwendung von unterschiedlichen Schrifttypen:

- Eigennamen oder besondere, z.B. fremdsprachliche Begriffe werden in *schräger* Schrift gesetzt. Beispiel: *IBJXodor*.
- Symbolische Parameter, die durch einen Wert zu ersetzen sind, werden *kursiv* geschrieben. Beispiel: *Option*.
- Text, der genau so, wie er dargestellt ist, eingegeben werden muß oder auf dem Bildschirm oder in einem File erscheint, ist in **Schreibmaschinenschrift** gesetzt. Beispielsweise der Aufruf von *IBJXodor* über die Tastatur: `IBJXodor x/fahne -v2`.

Die Programme sind in ANSI-C und in Java geschrieben und vermeiden bei der Installation jeden Eingriff in das Betriebssystem des Anwenders. Insbesondere werden keine Systemprogramme ausgetauscht.

*IBJmepod* und *IBJXodor* benötigen jeweils einen Lizenz-File (`*.licence`), der im gleichen Verzeichnis stehen muß wie das Programm. Er enthält den Namen des Anwenders und evtl. Hinweise auf Nutzungsrechte. Es handelt sich zwar um reine Text-Files, jede Veränderung des Inhalts macht sie aber unbrauchbar, so daß das jeweilige Programm nicht mehr lauffähig ist.

Falls Sie Probleme bei der Installation oder der Anwendung der Programme haben, senden Sie uns bitte eine Email nach [support@janicke.de](mailto:support@janicke.de). Wir helfen Ihnen gerne, können das aber nur, wenn wir Ihr Problem nachvollziehen können. Senden Sie uns also alle Informationen, die notwendig sind, um Ihr Problem bei uns zu rekonstruieren, insbesondere also die Eingabe-, Protokoll- und Ergebnis-Files.



## 2 Installation

Das Programmsystem IBJmepod wird auf einer CD ausgeliefert, die neben den eigentlichen Programmen *IBJmepod* und *IBJXodor* auch Hilfsprogramme, Beispieldateien und die Dokumentation enthält. Die CD enthält folgende Verzeichnisse:

### Acrobat

Der *Acrobat Reader* von Adobe, mit dem elektronische Dokumente (\*.pdf) gelesen werden können (Version 4.05).

### Jre

Das *Java Runtime Environment* (JRE) von SUN für die Benutzeroberfläche (Version 1.3.0).

### Mepod

Das Programmsystem MEPOD mit der Benutzeroberfläche *IBJmepod*. Es gibt folgende Unterverzeichnisse:

|              |                                      |
|--------------|--------------------------------------|
| doc          | Dokumentation in Form von PDF-Files. |
| IBJXodor-1.8 | Das Programm <i>IBJXodor</i> .       |
| x            | Beispielrechnungen                   |

### Archive

Eingabedaten und Ergebnisse zu Rechnungen, die im MEPOD-Handbuch aufgeführt sind (z.B. die Verifizierungs- und die Validierungsrechnungen). Zusätzlich stehen hier fertig installierte Versionen des *Acrobat Readers* und des JRE, die direkt von der CD aus lauffähig sind.

Zur Installation sind folgende Schritte nötig:

1. *Acrobat Reader* installieren.  
Rufen Sie hierzu das Programm auf, das im Verzeichnis *AcroRD* steht (Doppelklick) und folgen Sie den Anweisungen des Installationsprogramms. Falls der *Acrobat Reader* bei Ihnen schon installiert ist, können Sie diesen Schritt überschlagen.<sup>1</sup>
2. Das *Java Runtime Environment* installieren.  
Rufen Sie hierzu das Programm auf, das im Verzeichnis *Jre* steht (Doppelklick), und folgen Sie den Anweisungen des Installationsprogramms. Falls das JRE bereits bei Ihnen installiert ist, können Sie diesen Schritt überschlagen.

---

<sup>1</sup>Falls bei Ihnen Version 4.0 installiert ist und die Seite 11 des Handbuchs zu MEPOD (... \Mepod\doc\mepod.pdf) nicht richtig ausgedruckt wird (z.B. nur die obere Hälfte von großen Klammern), dann können Sie auch direkt die Version 4.05 auf der CD aufrufen (Archive\AcroRD\Reader\AcrRd32.exe).



### 3. Installation von *IBJmepod*.

Kopieren Sie hierzu das komplette Verzeichnis *Mepod* von der CD auf die Festplatte an eine Ihnen passende Stelle. Im Folgenden wird davon ausgegangen, daß es im Wurzelverzeichnis steht. Falls Sie es in ein anderes Verzeichnis kopiert haben, sind die im Folgenden verwendeten Pfadangaben entsprechend zu ändern.

Wenn das JRE installiert ist, wird *IBJmepod* durch einen Doppelklick auf den File *Mepod\IBJmepod.jar* gestartet. *IBJmepod* erwartet das Programm *IBJXodor* in einem Unterverzeichnis, dessen Name mit *IBJXodor* beginnt. Durch Anhängen der Versionsnummer können Sie zwischen verschiedenen Versionen von *IBJXodor* unterscheiden, die Ihnen von *IBJmepod* zur Auswahl angeboten werden. Der Programmname ist *wIBJXodor.exe*.

Falls Sie ohne Benutzeroberfläche mit *IBJXodor* arbeiten wollen, können Sie das Programm *Mepod\IBJXodor-1.8\IBJXodor.exe* in einem DOS-Fenster von einem passenden Verzeichnis aus auch direkt aufrufen.<sup>2</sup>

## 3 Die Benutzeroberfläche *IBJmepod*

Zur Zeit steht erst eine Kurzanleitung zur Verfügung. Für weitere Informationen wird auf die online-Hilfe hingewiesen (Menupunkt Info/Hilfe).

Das Programm *IBJmepod* dient dazu, die Eingabedatei für *IBJXodor* zu erstellen, die Rechnung zu starten und die Ergebnisse anzuzeigen. Es wird durch einen Doppelklick auf den File *Mepod\IBJmepod.jar* gestartet. Das Verzeichnis, in dem *IBJmepod.jar* steht, also *Mepod*, wird im folgenden als Stammverzeichnis bezeichnet.

### 3.1 Erstellung des Eingabe-Files

Nach dem Start wird in einem Fenster eine Tabelle mit einer Standard-Eingabedatei angeboten. Es ist die Datei *IBJXodor.input*, die im gleichen Verzeichnis steht wie das Programm *wIBJXodor.exe*. Sie ist in der Form, wie sie ausgeliefert wird, für eine Rechnung gedacht, bei der für eine punktförmige Quelle eine Jahresrechnung mit der Wetterstatistik *anonym.dat* durchgeführt werden soll. Durch Ändern von Werten (Doppelklick in der rechten Spalte) und Hinzufügen oder Löschen von Zeilen (über die rechte Maustaste erhalten Sie ein entsprechendes Popup-Menü) können Sie diese Tabelle Ihren Wünschen anpassen (siehe online-Hilfe).

Damit die Änderung eines Wertes vom Programm übernommen wird, müssen Sie nach der Änderung RETURN drücken!

<sup>2</sup>Die Programme *wIBJXodor.exe* und *IBJXodor.exe* sind funktional identisch, nur erwartet das letztere ein DOS-Fenster, welches das andere nicht benötigt.



Die fertige Tabelle muß in ein Arbeitsverzeichnis abgespeichert werden, das später auch die Rechenergebnisse aufnimmt. Sie können über **Projekt/Speichern als...** ein neues Verzeichnis anlegen (siehe online-Hilfe) oder ein bestehendes verwenden. Beim Speichern der Tabelle werden alle evtl. vorhandenen Rechenergebnisse in diesem Verzeichnis gelöscht!

## 3.2 Rechnung durchführen

Die Rechnung wird über **Start/Start** oder über das entsprechende Icon unter der Menu-Leiste gestartet. Während die Rechnung läuft, erscheint ein kleines Informations-Fenster und aus Sicherheitsgründen ist die weitere Arbeit mit *IBJmepod* gesperrt. Sie können aber im Ausgabe-Fenster den Fortschritt der Rechnung verfolgen: Hier wird aufgelistet, was *IBJXodor* bei einem Aufruf über das DOS-Fenster dort ausschreiben würde.

Vor dem Start der Rechnung werden im Arbeitsverzeichnis alle evtl. vorhandenen Ergebnisdateien gelöscht. Mit Ende der Rechnung verschwindet das kleine Informations-Fenster und Sie können wieder mit *IBJmepod* arbeiten und z.B. den Protokoll-File einsehen.

## 3.3 Ergebnisse darstellen

Im Menu-Punkt **Ergebnisse** können Sie auswählen, ob die Häufigkeit von Geruchsstunden (*frh.dmna*), die kumulative Geruchshäufigkeit (*fro.dmna*) oder die mittlere Geruchsstoffkonzentration (*cnc.dmna*) dargestellt werden soll. Die Karteikarte *Tabelle* gibt Ihnen eine tabellarische Darstellung, die Karteikarte *Grafik* eine grafische Darstellung in Form von Isolinien.

Die tabellarische Darstellung ist mit der im entsprechenden dmna-File identisch. Sie können die Werte aus der Tabelle in eine andere Anwendung übernehmen, indem Sie mit der Maus den interessierenden Bereich auswählen und mit *Ctrl-C* (bzw. *Strg-C*) in die Zwischenablage kopieren. Durch einen Doppelklick auf eine beliebige Zelle wird die gesamte Tabelle ausgewählt, ein Klick auf die linke obere Ecke löscht die Auswahl.

Über den Menu-Punkt **Zeichnen** im Popup-Menu (rechte Maustaste) gelangen Sie zu einer Darstellung in Form einer Rasterkarte. Hierbei sind die Felder quadratisch und mit einer Farbe unterlegt, die vom Zahlenwert in dieser Zelle abhängt. Die verwendete Farbtabelle können Sie in der Karte *Grafik* einsehen und verändern. Bei der Rasterkarte können Sie die Schriftgröße verändern und Sie können sie ausdrucken oder als PNG- oder BMP-File speichern. Falls die Tabelle nicht auf ein Blatt paßt, wird sie in mehrere kleinere Tabellen zerlegt.

Die grafische Darstellung wird auf der Karte *Grafik* gezeigt. Es werden immer 9 Isolinien gezeichnet, die auf Wunsch farbig ausgefüllt sind. Die Farbpalette und ihre Zuordnung zu Werten kann über *Palette...* im Popup-Menu geändert werden. Die



Farbfelder werden durch einen Doppelklick editierbar. Diese Änderungen können gespeichert und bei Bedarf auch wieder eingelesen werden.

Im Popup-Menü finden Sie auch die Möglichkeit, die Darstellung zu modifizieren und das Bild zu drucken oder als PNG- oder BMP-File zu speichern. Bei **Einstellungen...** wird Ihnen ein Fenster angeboten, in dem Sie die Größe des Zeichenfeldes, die Auflösung des Bildes (wird ebenfalls bei der Rasterkarte verwendet) und die Achseneinteilung ändern können. Im Feld *Überlagertes Bild:* können Sie den Namen eines HPGL-Files angeben, der dann in die Zeichnung übernommen wird. Hiermit können z.B. Quellorte, Grundstücksgrenzen, Straßen oder ähnliches eingeblendet werden (siehe Anhang B).



## 4 Das Programm *IBJXodor*

Das Modell MEPOD ist als Computer-Programm *IBJXodor* in der Programmiersprache ANSI-C realisiert. Es ist ein reines Text-Programm, läuft also unter Windows in einem DOS-Fenster. Das Programm ist Teil einer Reihe von Programmen zu Problemen der Stoffausbreitung in der Atmosphäre, die alle eine ähnliche Struktur besitzen, teilweise gleiche Module enthalten und das gleiche Datenformat verwenden.

Das Programm *IBJodor* ist eine eingeschränkte Version von *IBJXodor*, die frei zur Verfügung steht. Die bestehenden Einschränkungen sind im Folgenden jeweils angegeben. Hinweise, die sich auf das MEPOD-Handbuch beziehen, sind durch ein vorangestelltes „m.“ gekennzeichnet.

### 4.1 Übersicht

Das Programm arbeitet nicht interaktiv sondern verwendet einen Eingabe-File, den Command-File, in dem alle Aufgaben spezifiziert sind, die das Programm in einem Rechenlauf durchführen soll. Bei der Berechnung von Jahreskenngrößen wird noch die Wetterstatistik als separater File benötigt. Während der Rechnung schreibt das Programm ein Arbeitsprotokoll in den Protokoll-File und erstellt eventuell weitere Ausgabe-Files. Alle diese Files haben vorgegebene Namen und befinden sich in einem einzigen Verzeichnis, dem Arbeitsverzeichnis. Zum Aufruf des Programms ist einzugeben:

```
IBJXodor Arbeitsverzeichnis Option ...
```

Mögliche Optionen sind:

**-e :**

Alle im Command-File eingelesenen Textzeilen werden im Protokoll-File protokolliert.

**-h :**

Alle möglichen Aufgaben und Eingabeparameter werden am Bildschirm aufgelistet und anschließend das Programm beendet.

**-i Command :**

Der File mit dem Namen *Command* soll als Command-File verwendet werden (der Standardname ist *odrcmd.txt*). Diese Option darf nicht zusammen mit *IBJmepod* verwendet werden.

**-l Log :**

Der Protokoll-File soll den Namen *Log* haben (der Standardname ist *odrlog.txt*). Diese Option darf nicht zusammen mit *IBJmepod* verwendet werden.

**-q :**

Es werden keine Meldungen auf den Bildschirm geschrieben (*quiet*).



**-v Verbose :**

Mit *Verbose* wird angegeben, wie ausführlich das Programm Meldungen auf dem Bildschirm und im Protokoll-File ausgibt. Der Standardwert ist 0 (keine Ausgabe). Dieser Wert ist notwendig, wenn Ergebnisse in den Protokoll-File geschrieben werden und diese Auflistung nicht unterbrochen werden soll. Sonst sollte `-v1` verwendet werden.

Der Command-File (Standardname ist `odrcmd.txt`) ist ein reiner Text-File, kann also vom Benutzer mit einem Text-Editor erstellt werden. Der Protokoll-File (Standardname ist `odrlog.txt`) ist ebenfalls ein Text-File, der vom Programm bei jedem Programmlauf angelegt wird. Die Rechenergebnisse stehen entweder im Protokoll-File oder, wenn sie umfangreicher sind, in separaten Ausgabe-Files.

Der Command-File besteht aus mehreren Abschnitten. Jeder Abschnitt definiert eine Aufgabe mit den dazugehörigen Parametern. Eine vollständige Auflistung aller möglicher Aufgaben und der zugeordneten Parameter wird in Abschnitt 4.2 gegeben. Die erste Zeile eines Abschnittes beginnt mit dem Zeichen „\*“, unmittelbar gefolgt von einem Namen, und definiert die Teilaufgabe, die ausgeführt werden soll, beispielsweise

**\*Dimensionierung**

Signifikant ist hierbei nur der erste Buchstabe, also das „D“, wobei auch nicht zwischen Klein- und Großschreibung unterschieden wird. Genauso gut hätte man also auch schreiben können

**\*dimensions**

Es folgen in diesem Falle die Parameter, die zur Dimensionierung der Felder benötigt werden, also z.B.

```
*Dimensionierung      ' kleines Testnetz
Nz    10              ' Das vertikale Netz hat 10 Intervalle
Nq    10              ' 10 Quellen werden später definiert
Mp    20              ' Es werden maximal 20 Aufpunkte verwendet
```

Bei den Parameternamen sind nur die ersten beiden Zeichen signifikant und es wird ebenfalls nicht zwischen Klein- und Großschreibung unterschieden. Anschließend folgen, durch Leerzeichen und/oder Tabulatoren getrennt, der oder die Werte. Dabei kann es sich um ganze Zahlen, Gleitkommazahlen oder Zeichenketten handeln. Zeichenketten können in Hochkommata eingeschlossen sein. Auch die Zeile, welche die Definition der Aufgabe enthält, kann nach dem Aufgabennamen noch Parameterwerte enthalten.

Ein Apostroph leitet einen Kommentar ein, der beim Einlesen überschlagen wird. Beginnt eine Zeile mit einem Leerzeichen oder einem Minuszeichen, wird die ganze Zeile als Kommentar angesehen.

Die erste Aufgabe muß immer **\*D** sein, denn nachdem die erste Aufgabe eingelesen ist, wird die Dimensionierung der internen Felder vorgenommen, die später nicht mehr geändert werden kann. Dies gilt auch für Eingabeparameter, für die mehr als



ein Wert anzugeben ist. Wenn die Anzahl vom Benutzer festgelegt werden kann, existiert immer ein Parameter im Abschnitt *\*D*, der für die Dimensionierung verwendet wird. Beispielsweise kann der Benutzer festlegen, wieviele Intervalle  $N_z$  das vertikale Raster  $z_k$  besitzen soll, das durch den Eingabeparameter *Zk* im Abschnitt *\*P* definiert wird. Die Anzahl der Gitterpunkte ist um 1 größer, also sind für *Zk*  $N_z + 1$  Werte anzugeben:

```
*Dimensionierung      ' kleines Testnetz
Nz   10                ' Das vertikale Netz hat 10 Intervalle
*Parameter            ' Weitere Parameter
Zk   0  5 10 15 20 25 32 40 50 70 100      ' Nz+1 Werte
```

Die letzte Aufgabe sollte immer *\*Ende* sein, damit das Programm definiert beendet wird.

## 4.2 Eingabe-Parameter

Das Programm *IBJXodor* erkennt folgende Aufgaben:

### *\*A*

Es werden Aufpunkte, an denen die Konzentration und die Geruchshäufigkeit berechnet werden soll, festgelegt. Die Höhe der Aufpunkte über Grund wird durch den Parameter *Zp* im Abschnitt *\*D* für alle Aufpunkte einheitlich festgelegt. Die berechneten Werte werden mit der Aufgabe *\*L* im Protokoll-File aufgelistet.

### *\*C*

Es werden intern alle Ergebnistabellen gelöscht, so daß anschließend eine weitere Rechnung durchgeführt werden kann.

### *\*D*

Die Dimensionen aller verwendeten Felder werden festgelegt. Diese Aufgabe darf nur einmal definiert werden und muß als erste im Command-File erscheinen. Zusätzlich werden Parameter festgelegt, die nur einmal gesetzt werden dürfen und dann innerhalb eines Rechenlaufes unverändert bleiben müssen.

### *\*E*

Der Rechenlauf wird beendet. Es werden anschließend keine Eingabedaten mehr eingelesen.

### *\*G*

Das Rechengitter, auf dem Konzentrationswerte und Geruchshäufigkeiten berechnet werden sollen, wird festgelegt. Dies geschieht nur einmal zu Beginn einer Rechnung. Die Höhe der Aufpunkte über Grund wird durch den Parameter *Zp* im Abschnitt *\*D* festgelegt. Falls nur einzelne Aufpunkte verwendet werden (Aufgabe *\*A*), kann dieser Abschnitt auch fehlen. Die berechneten Werte werden mit der Aufgabe *\*S* als separate Files ausgeschrieben.

**\*I<sup>3</sup> File**

Der File mit dem Namen *File* wird an dieser Stelle eingefügt. Er kann mehrere Abschnitte mit zugehörigen Parameterwerten enthalten. Eine Tilde unmittelbar vor dem Filenamen bedeutet, daß eine Pfadangabe in *File* nicht vom aktuellen Verzeichnis (Stammverzeichnis) sondern vom Arbeitsverzeichnis aus gilt.

**\*L [plm | pnt | prf]**

Im Protokoll-File werden die Fahnenfunktionen bezogen auf die Quellstärke 1 (**plm**), die Werte an den Aufpunkten (**pnt**) oder die meteorologischen Profilkfunktionen (**prf**) aufgelistet. Die Tabellen haben folgende Spalten:

|            |                            |  |
|------------|----------------------------|--|
| <b>plm</b> | : Abstand                  | Quellabstand $x$ (m)   |
|            | <b>Cy(z=0)</b>             | Integrierte Konzentration $C_y$ am Erdboden bei $z = 0$ (GE/m <sup>2</sup> )                               |
|            | <b>Cy(z=z<sub>p</sub>)</b> | Integrierte Konzentration $C_y$ in Höhe der Aufpunkte bei $z = z_p$ (GE/m <sup>2</sup> )                   |
|            | <b>Cy(z=h)</b>             | Integrierte Konzentration $C_y$ in mittlerer Quellhöhe bei $z = h_q + \frac{1}{2}c_q$ (GE/m <sup>2</sup> ) |
|            | <b>Sy</b>                  | Gesamte Fahnenbreite $\sigma_y$ (m)  |
|            | <b>Sd</b>                  | Breite einer Kernfahne $\hat{\sigma}_y$ (m)  |
|            | <b>Cm(z=z<sub>p</sub>)</b> | Konzentration unter der Fahnenachse bei $z = z_p$ (GE/m <sup>3</sup> )                                     |
|            | <b>Fluß</b>                | Insgesamt transportierte Menge $\int C_y(z) u(z) dz$ (GE/s)  |
| <b>pnt</b> | : <b>i</b>                 | Index des Aufpunktes (bei 1 beginnend)   |
|            | <b>Xp</b>                  | $x$ -Koordinate des Aufpunktes (m)   |
|            | <b>Yp</b>                  | $y$ -Koordinate des Aufpunktes (m)   |
|            | <b>Cnc</b>                 | Mittlere Konzentration (GE/m <sup>3</sup> )  |
|            | <b>Frh</b>                 | Häufigkeit von Belästigungsstunden   |
|            | <b>Fro</b>                 | Kumulative Geruchshäufigkeit   |
| <b>prf</b> | : <b>Z</b>                 | Höhe $z_k$ über Grund (m)  |
|            | <b>Ux</b>                  | Windgeschwindigkeit $u_k$ (m/s)  |
|            | <b>Kz</b>                  | Vertikaler Austauschkoefizient $K_{zz;k}$ (m <sup>2</sup> /s)  |
|            | <b>Sw</b>                  | Vertikale Windfluktuation $\sigma_{w;k}$ (m/s)   |

**\*P [run]**

Die für die Ausbreitungsrechnung benötigten Parameter (Meteorologie, Stoffdaten) werden festgelegt. Diese Aufgabe kann beliebig innerhalb einer Rechnung auftreten. Ist der Parameter **run** angegeben, dann werden die Fahnen neu berechnet und ihre Beiträge zu den Werten an den Aufpunkten addiert.

**\*Q**

Die Quelldaten (Position, Ausdehnung, Quellstärke) werden festgelegt. Diese Aufgabe kann beliebig innerhalb einer Rechnung auftreten, so daß Quellort und/oder Quellstärke variabel sein können.

**\*S [cnc | fro | frh]**

Die auf dem Rechengitter berechneten Größen werden als separate Files beschrieben. Als Optionen können angegeben werden:

---

<sup>3</sup>Diese Aufgabe steht in *IBJodor* nicht zur Verfügung



**cnc** Mittlere Konzentration (Filename **cnc.dmna**)  
**frh** Häufigkeit von Geruchsstunden (Filename **frh.dmna**)  
**fro** Kumulative Geruchshäufigkeit (Filename **fro.dmna**)

Die Struktur dieser Files ist in Anhang A beschrieben. Die Ausgabe kann auch kombiniert erfolgen, z.B. mit der Option **cnc+frh+fro**.

**\*T** [*Auswahl*]

Es werden die im Augenblick gültigen Parameterwerte für alle Abschnitte, die in *Auswahl* aufgeführt sind, in den Protokoll-File geschrieben. Ist *Auswahl* nicht angegeben, dann werden alle Abschnitte genommen. Dies ist äquivalent mit der Angabe von „\*a\*d\*g\*1\*p\*q\*s“.

**\*W<sup>4</sup>** *File*

Es wird die Wetterstatistik mit dem Filenamen *File* abgearbeitet. Eine Tilde unmittelbar vor dem Filenamen bedeutet, daß eine Pfadangabe in *File* nicht vom aktuellen Verzeichnis (Stammverzeichnis) sondern vom Arbeitsverzeichnis aus gilt. Die Statistik kann im TA-Luft-Format oder im alten DWD-Format vorliegen. Diese Aufgabe ist äquivalent mit einer Folge von Aufgaben „\*P run“, bei denen jeweils Windgeschwindigkeit *Ua*, Monin-Obukhov-Länge *Lm*, Mischungsschichthöhe *Hm* und die relative Häufigkeit der Windrichtungssektoren *Dw* neu gesetzt werden.

**\*\*** *Text*

Der Text *Text* wird in den Protokoll-File geschrieben, wobei einschließende Doppelhochkommata entfernt werden. Auf diese Weise können bestimmte Abschnitte im Protokoll-File besser kenntlich gemacht werden.

Die einzelnen Eingabeparameter sind diesen Aufgaben zugeordnet, dürfen also nur in dem betreffenden Abschnitt gesetzt werden. Sie stellen entweder ganze Zahlen, Gleitkommazahlen oder Zeichenketten dar und werden zu Beginn der Rechnung mit einem Standardwert belegt. Sie behalten ihren Wert solange bei, bis er durch Eingabe eines neuen Wertes geändert wird. Ausnahmen hiervon sind explizit aufgeführt. Alle Längen sind in Metern anzugeben, alle Zeiten in Sekunden.

Die einem Parameter zugewiesenen Werte folgen dem Namen, von diesem und untereinander durch Leerzeichen und/oder Tabulatoren getrennt. Zeichenketten müssen, wenn sie nicht-alphanumerische Zeichen enthalten, in Hochkommata eingeschlossen sein.

In der folgenden Tabelle steht für jeden Parameter in einer Zeile der *Name*, anschließend der *Datentyp* (*integer*, *float* oder *string*), dahinter in Klammern die *Anzahl* der Werte und schließlich die *Standardsetzung*. In den darauf folgenden Zeilen ist die Bedeutung und Handhabung dieses Parameters beschrieben.

Angaben, die nicht auf *IBJodor* zutreffen, sind durch ein hochgestelltes Kreuz (†) gekennzeichnet.

---

<sup>4</sup>Diese Aufgabe steht in *IBJodor* nicht zur Verfügung.



- \*A** Aufpunkte \_\_\_\_\_
- np** *integer*(1) 0  
Tatsächliche Anzahl der Aufpunkte. **np** darf höchstens gleich dem Wert von **mp** aus dem Abschnitt **\*D** sein. Es werden vom Programm nur die Werte für die ersten **np** Punkte berechnet, die anderen werden ignoriert und auch im Protokoll-File nicht aufgelistet. Die Verwendung von **np** ist sinnvoll, wenn in einem Rechenlauf mehrere Fahnenbegehungen mit unterschiedlicher Probandenzahl nachgerechnet werden sollen.
- xp** *float*(**mp**) 0  
 $x$ -Koordinate des Aufpunktes.
- yp** *float*(**mp**) 0  
 $y$ -Koordinate des Aufpunktes.
- \*C** Clear \_\_\_\_\_
- \*D** Dimensionierung \_\_\_\_\_
- dz** *float*(1) 0  
Intervallgröße im vertikalen Netz. Ist ein Wert größer null angegeben, dann wird ein äquidistantes Netz mit **nz** Intervallen erzeugt. Hat **dz** einen Wert kleiner gleich null, dann wird ein exponentielles Netz gemäß Gleichung (m.60) mit **nz** Intervallen,  $A_z = 80$  und  $z_{N_z} = 1100$  erzeugt.† Dieses Netz kann mit dem Parameter **zk** im Abschnitt **\*P** überschrieben werden.
- l1** *integer*(1) -10  
Unterer Indexwert  $l_{\min}$  bei der Definition des  $c$ -Rasters nach Gleichung (m.61).
- mc** *integer*(1) 4000  
Maximale Anzahl von Bytes, die in einer Eingabezeile vorkommen können.
- me** *string*(1) "GE"  
Verwendete Mengeneinheit für den freigesetzten Stoff.
- mp** *integer*(1) 0  
Maximal mögliche Anzahl von Aufpunkten. Die tatsächlich verwendete Anzahl von Aufpunkten kann mit dem Parameter **np** im Abschnitt **\*A** angegeben werden.
- nc** *integer*(1) 5  
Anzahl von Intervallen pro Dekade  $A_c$  bei der Definition des  $c$ -Rasters nach Gleichung (m.61).
- n1** *integer*(1) 20  
Anzahl der Intervalle  $l_{\max} - l_{\min}$  im  $c$ -Raster nach Gleichung (m.61).
- nm** *integer*(1) 9  
Anzahl der Kernfahnen  $\tilde{N}$  für die halbe Gesamtfahne ( $\eta_{\max}$  hat standardmäßig den Wert 3). Ist **nm** null, dann wird ohne Mäandern gerechnet und die Breite der Kernfahne  $\hat{\sigma}_y$  wird gleich der Breite der Gesamtfahne  $\sigma_y$  gesetzt.
- nq** *integer*(1) 1  
Anzahl der Quellen.†



**nw** *integer*(1) 36

Anzahl der Windrichtungssektoren in der Wetterstatistik. Sollen nur Einzelsituationen gerechnet werden (z.B. bei Fahnenbegehungen), dann ist **nw** auf 1 und ebenfalls **nu** in Abschnitt \*P auf 1 zu setzen.

**nz** *integer*(1) 200

Anzahl der Intervalle  $N_z$  im vertikalen Netz.

**op** *string*(1) ""

Mit der Zeichenkette **op** können zusätzliche Optionen für die Verarbeitung angegeben werden. Folgende Optionen können in **op** enthalten sein:

**intern** Es werden weitere interne Parameter zugänglich gemacht, die zur Kontrolle aufgelistet oder auch modifiziert werden können.

**onlymean** †Es wird nur die mittlere Konzentration berechnet, ohne Konzentrationsfluktuationen oder Mäandern zu betrachten. Dies impliziert, daß **nm** auf null gesetzt wird.

**logxA<sub>x</sub>** † Es wird eine exponentielle Intervallunterteilung in  $x$ -Richtung gewählt (siehe Gleichung m.59) und der Parameter  $A_x$  gesetzt. Sein Wert ist unmittelbar an **logx** anzuhängen, standardmäßig hat er den Wert 400. Mit der Standardsetzung von **li** in Abschnitt \*Q wird automatisch ein exponentielles Netz erzeugt, so daß diese Option nur gesetzt zu werden braucht, wenn der Parameter  $A_x$  verändert werden soll.

**logzA<sub>z</sub>** † Es wird eine exponentielle Intervallunterteilung in  $z$ -Richtung gewählt (siehe Gleichung m.60) und der Parameter  $A_z$  gesetzt. Sein Wert ist unmittelbar an **logz** anzuhängen, standardmäßig hat er den Wert 200. Mit der Standardsetzung von **dz** in Abschnitt \*D wird automatisch ein exponentielles Netz erzeugt, so daß diese Option nur gesetzt zu werden braucht, wenn der Parameter  $A_z$  verändert werden soll.

**ti** *string*(1) "TEST"

Kennzeichnung zur Dokumentation des Rechenlaufes.

**xf** *float*(1) 0

Rechtswert des Koordinatenursprungs (Referenzpunkt).

**yf** *float*(1) 0

Hochwert des Koordinatenursprungs (Referenzpunkt).

**zp** *float*(1) 1.5

Höhe der Aufpunkte über Grund. Dies betrifft sowohl die in Abschnitt \*A definierte Liste von Aufpunkten wie auch das in Abschnitt \*G definierte Gitter von Aufpunkten.

**\*E** Ende \_\_\_\_\_

**\*G** Gitter \_\_\_\_\_

**dr** *float*(1) 10

Horizontale Maschenweite des Aufpunktgitters.



- nx** *integer*(1) 0  
Anzahl der Gitterpunkte in  $x$ -Richtung.
- ny** *integer*(1) 0  
Anzahl der Gitterpunkte in  $y$ -Richtung.
- xr** *float*(1) 0  
 $x$ -Koordinate des linken (westlichen) Randes.
- yr** *float*(1) 0  
 $y$ -Koordinate des unteren (südlichen) Randes.
- \*I†** Include *File* \_\_\_\_\_
- \*L** Liste *Name+Name+...* \_\_\_\_\_
- nd** *integer*(1) 0  
Anzahl der auszulassenden Punkte für *Name=plm* oder *Name=prf*. Hat **nd** beispielsweise den Wert 1, dann wird nur jeder zweite Punkt aufgelistet.
- fc** *float*(1) 1  
Faktor, mit dem Konzentrationswerte bei der Auflistung multipliziert werden.
- \*P** Parameter [**run**] \_\_\_\_\_
- a1** *float*(1) 0  
Faktor  $\alpha$  im Differenzenverfahren, siehe Gleichung (m.57). Hat **a1** den Wert null, dann wird voll implizit gerechnet. Sonst werden bei Bedarf Zwischenschritte eingefügt, um die Stabilitätsbedingung (m.58) zu erfüllen.
- dw** *float*(**nw**) 1  
Relative Häufigkeit der Windrichtungen aus den **nw** Sektoren. Wird **dw** nicht gesetzt, dann haben alle Windrichtungen die gleiche Häufigkeit, es wird also mit einer isotropen Windrose gerechnet.
- ga** *float*(1) 0.1  
Kritischer Zeitanteil mit Geruch ( $\kappa_{\min;0}$ ). Wird dieser Zeitanteil innerhalb einer Stunde überschritten, dann zählt diese Stunde als Geruchsstunde.
- gs** *float*(1) 1  
Feldgeruchsschwelle  $c_{F;0}$  in  $\text{GE}/\text{m}^3$ .
- ha** *float*(1) 10  
Anemometerhöhe  $h_a$  über Grund.
- hm** *float*(1) 0  
Mischungsschichthöhe  $h_m$ . Wird kein Wert angegeben, dann wird für  $h_m$  ein der Stabilitätsklasse gemäß VDI 3782/1 zugeordneter Wert verwendet. Ist dies nicht möglich oder ist der Wert null angegeben, dann wird  $h_m$  intern vom Grenzschichtmodell berechnet.
- kl†** *string*(1) "III/1"  
Stabilitätsklasse nach Klug/Manier. Angabe in der Form: "I", "II", "III/1", ... oder: "1", "2", "3.1", ... oder: "F", "E", "D", ... . Sind **lm** oder **hm** explizit angegeben, dann wird für den betreffenden Parameter der Wert von **kl** ignoriert.



- kw** *float*(1) 2.5  
Weber-Fechner-Koeffizient  $k_W$ .
- ld** *float*(1) 0.5  
Auflösungsvermögen  $\lambda$  in der Wahrnehmungsfunktion, siehe Gleichung (m.67).
- lm** *float*(1) 0  
Monin-Obukhov-Länge  $L_M$ . Wird kein Wert angegeben, dann wird — sofern möglich — ein der Stabilitätsklasse zugeordneter Wert verwendet (s. Beschreibung des Grenzschichtmodells). Hat **lm** den Wert null, dann wird mit indifferenter Schichtung gerechnet.
- mf** *float*(1) -1  
Faktor  $\tilde{f} = \tilde{\sigma}_y / \hat{\sigma}_y$ , der die Stärke des Mäanderns beschreibt. Wird ein Wert größer null angegeben, dann wird  $\hat{\sigma}_y = \sigma_y / \sqrt{1 + \tilde{f}^2}$  gesetzt.
- mx** *integer*(1) 0  
Schrittweite in  $x$ -Richtung bei der Ausgabe des Konzentrationsprofils. Ist ein Wert größer null angegeben, dann wird bei der Berechnung der  $c_{ik'}$  jeweils nach Fortschreiten um **mx** Schritte in  $x$ -Richtung ein Vertikalprofil der Konzentration in den Protokoll-File geschrieben.
- mz** *integer*(1) 1  
Schrittweite in  $z$ -Richtung bei der Ausgabe des Konzentrationsprofils, sofern **mx** einen Wert größer als null hat.
- nu** *integer*(1) 5  
Anzahl der Untersektoren pro Windrichtungssektor beim Durchrechnen einer Windrose. Rechnet man mit 36 Windrichtungssektoren (Parameter **nw** im Abschnitt \*D), dann wird jeder Sektor als 5 Untersektoren der Breite 2 Grad und gleicher Häufigkeit gerechnet. Beim Rechnen von Einzelsituationen (Fahnenbegehung) ist **nu** ebenso wie **nw** auf den Wert 1 zu setzen.
- pd** *float*(1) 2  
Parameter  $\hat{\nu}$  der Weibull-Verteilung.
- re** *float*(1) 10  
Richtung der Mitte des ersten Sektors der Windrose. Die Windrose wird von Nord aus im Uhrzeiger gezählt. Nach Konvention des DWD liegt der erste Sektor zwischen 5 Grad und 15 Grad.
- sv** *float*(1) 0  
Horizontale Windfluktuation  $\sigma_v$ . Wird kein Wert angegeben, dann wird  $\sigma_v$  aus dem Grenzschichtmodell übernommen.
- sw** *float*(1) 0  
Vertikale Windfluktuation  $\sigma_w$  in Anemometerhöhe  $h_a$ . Wird kein Wert angegeben, dann wird  $\sigma_w$  aus dem Grenzschichtmodell übernommen.
- tm** *float*(1) 3600  
Mittelungszeit  $\bar{t}$  für die Berechnung der Grenzschichtprofile. Üblicherweise ist der Bezugszeitraum 1 Stunde, beim Nachrechnen von Fahnenbegehungen ist hier die tatsächliche Begehungszeit einzusetzen.



- ua** *float*(1) 3  
Windgeschwindigkeit  $u_a$  in Anemometerhöhe  $h_a$ .
- us** *float*(1) 0  
Schubspannungsgeschwindigkeit  $u_*$ . Ist kein Wert größer null angegeben, dann wird  $u_*$  vom Grenzschichtmodell aus dem Windprofil berechnet.
- z0** *float*(1) 0.5  
Rauhigkeitslänge  $z_0$ .
- zk** *float*(nz+1) 0  
Vertikales Raster  $z_k$ . Ist **zk** nicht angegeben, dann wird das vertikale Netz automatisch aus den Parametern **nz**, **dz** und evtl.  $A_z$  berechnet.
- 
- \*Q** Quellen
- aq** *float*(nq) 0  
Horizontale Ausdehnung der Quelle in  $x$ -Richtung (vor der Drehung).
- bq** *float*(nq) 0  
Horizontale Ausdehnung der Quelle in  $y$ -Richtung (vor der Drehung).
- cq** *float*(nq) 0  
Vertikale Ausdehnung der Quelle.
- dq** *float*(nq) 0  
Durchmesser der Quelle (wird nur zur Berechnung der Abgasfahnenüberhöhung nach VDI 3782/3 verwendet).†
- eq** *float*(nq) 1  
Quellstärke in GE/s.
- hq** *float*(nq) 10  
Höhe der Quelle (Unterkante bei vertikaler Ausdehnung).
- hu** *float*(nq) 0  
Überhöhung der Abgasfahne. Nur wenn dieser Parameter nicht angegeben ist, werden die Parameter **dq**, **vq** und **tq** oder **qq** zur Berechnung einer Abgasfahnenüberhöhung nach VDI 3782/3 wirksam.†
- li** *float*(nq) -3.5  
Maschenweite des Rasters zur Berechnung der Fahnenfunktionen in  $x$ -Richtung. Ist ein negativer Wert angegeben, dann wird ein exponentielles Raster nach Gleichung (m.59) erzeugt und dabei  $l_x$  gleich dem Absolutwert von **li** gesetzt.†
- lq** *float*(nq) -1  
Maximale Länge  $L_q$  bei der Aufrasterung einer Flächenquelle. Ist **lq** nicht gesetzt, wird es gleich der halben mittleren Höhe der Quelle gesetzt. Wird es auf Werte kleiner gleich null gesetzt, dann wird **lq** abhängig von der Entfernung des Aufpunktes gewählt.†
- ni** *integer*(nq) 0  
Anzahl der Intervalle  $N_x$  in  $x$ -Richtung bei der Berechnung der Fahnenfunktionen. Ist der Wert von **ni** kleiner gleich null, dann wird  $N_x$  so groß gewählt, daß die berechneten Fahnen alle Aufpunkte erreichen.†



- qq** *float*(nq) 0  
Emittierter Wärmestrom in MW (zur Berechnung der Abgasfahnenüberhöhung nach VDI 3782/3).†
- rq** *float*(nq) 0  
Drehwinkel der rechteckigen Quelle um den linken (westlichen) unteren (südlichen) Eckpunkt. Er läuft gegen den Uhrzeigersinn und ist der Winkel zwischen der mit **aq** bezeichneten Seite und der positiven *x*-Achse.
- tq** *float*(nq) 10  
Abgastemperatur in Grad Celsius (zur Berechnung der Abgasfahnenüberhöhung nach VDI 3782/3, falls **qq** den Wert 0 hat).†
- xq** *float*(nq) 0  
*x*-Koordinate des linken (westlichen) unteren (südlichen) Eckpunktes (vor der Drehung um den Winkel **rq**).
- yq** *float*(nq) 0  
*y*-Koordinate des linken (westlichen) unteren (südlichen) Eckpunktes (vor der Drehung um den Winkel **rq**).
- \*S** Speichern *Name+Name+...* \_\_\_\_\_
- fa** *float*(1) 1  
Faktor zu Darstellung der Werte der betr. Variablen in einem Text-File. Bevor die Werte im Format **fo** ausgeschrieben werden, werden sie mit **fa** multipliziert.
- fi** *string*(1) ""  
File-Name, der zum Ausschreiben der Variablen *Name* verwendet werden soll. Der Wert von **fi** bleibt nicht erhalten, er muß also bei jeder Aufgabe *Speichern* neu angegeben werden. Ist **fi** nicht angegeben, dann wird als File-Name *Name.dmn* genommen.
- fo** *string*(1) "%12.4e"  
Format zur Darstellung der Daten in einem Text-File (siehe Anhang A).
- id** *string*(1) "TEST"  
Zeichenkette zur Identifizierung der Daten für Dokumentationszwecke.
- mo** *string*(1) "text"  
Modus der Datendarstellung. Wird **binary** angegeben, dann werden nur die Strukturinformationen in den Text-File geschrieben, die Daten aber in einen separaten Binär-File (nicht bei *IBJmepod* verwenden!). Er hat den gleichen Namen wie der Text-File, aber die Namensweiterung **dmnb**.
- se** *string*(1) ""  
Auswahl der auszuschiebenden Daten (siehe Anhang A). Wird **se** nicht angegeben, dann werden die Daten in der intern gespeicherten Reihenfolge geschrieben (dies entspricht "*i+:j+*" bei einem 2-dimensionalen Feld).
- \*W†** Wetterstatistik *File* \_\_\_\_\_



## 4.3 Beispiele

Im folgenden werden zwei Beispiele vorgeführt. Im ersten wird eine Fahnenbegehung nachgerechnet, wobei die Geruchshäufigkeiten an verschiedenen Aufpunkten ausgerechnet werden. Im zweiten Beispiel wird eine Jahresstatistik durchgerechnet und mittlere Konzentration, Geruchshäufigkeit und Geruchsstundenhäufigkeit als Punktwerte auf einem Raster berechnet.

### 4.3.1 Fahnenbegehung

Die Quelle besteht aus zwei Ställen, die als Volumenquellen behandelt werden, da die Freisetzung über ein ganzes Areal von Dachreitern erfolgt und die Ställe so ausgedehnt sind, daß mit einem Einmischen der Abluft in den Leewirbel gerechnet werden muß. Der Eingabe-File `odrcmd.txt` im Arbeitsverzeichnis `x/fahne` sieht folgendermaßen aus:<sup>5</sup>

```
*d          'DIMENSIONIERUNG VON FELDERN UND RECHENGEBIET
nq          2          'Anzahl der Quellen
mp          12         'Maximale Anzahl der Aufpunkte
nw          1          'Anzahl der Windrichtungen
zp          1.5        'Z-Koordinate der Aufpunkte (m)
*a          'DEFINITION DER AUFPUNKTE
np          12         'Anzahl der Aufpunkte
xp 1079.0 1004.0 929.0 854.0 779.0 705.0 630.0 555.0 480.0 405.0 331.0 256.0
yp 922.0 916.0 911.0 906.0 901.0 895.0 890.0 885.0 880.0 875.0 869.0 864.0
*q          'QUELLEN: LAGE, AUSDEHNUNG UND QUELLSTÄRKEN
xq          41         -19      'X-Koordinate der Quelle/Eckpunkt (m)
yq          -41        -25      'Y-Koordinate der Quelle/Eckpunkt (m)
aq          58.0       58.0     'Horizontale Ausdehnung der Quelle (m)
bq          38.0       38.0     'Horizontale Ausdehnung der Quelle (m)
rq          75.0       75.0     'Winkel von aq gegen Ost (Grad gegen den Uhrzeiger)
hq          0.0        0.0     'Höhe der Quelle/Unterkante (m)
cq          5.0        5.0     'Vertikale Ausdehnung der Quelle (m)
eq 55200.0 55200.0      'Quellstärke (GE/s)
lq          5.0        5.0     'Maximale Rasterweite einer Flächenquelle (m)
*p          ' [RUN] PARAMETER DER AUSBREITUNGSRECHNUNG
nu          1          'Anzahl der Unter-Sektoren
ha          10.0       'Höhe des Anemometers über Grund (m)
tm          600        'Mittelungszeit für die Fahne (s)
z0          0.0330     'Rauhigkeitslänge (m)
gs          0.50       'Feld-Geruchsschwelle (GE/m3)
*p run      ' [RUN] PARAMETER DER AUSBREITUNGSRECHNUNG
dw          600        'Dauer der Windrichtung (h)
re          220        'Richtung des ersten Sektors (Grad gegen Nord im Uhrzeiger)
ua          6.44       'Windgeschwindigkeit in Anemometerhöhe (m/s)
lm          -387.6     'Monin-Obukhov-Länge (m)
hm          800.0     'Höhe der Mischungsschicht (m)
sv          0.780     'Sigma-V (m/s)
sw          0.420     'Sigma-W (m/s)
** "## 24.08.92 10:00" 'KOMMENTAR FÜR DIE LOG-DATEI
*l pnt      '[PLM|PRF|PNT] PROFILE AUFLISTEN
*e          'ENDE DER RECHNUNG
```

Die Feldgeruchsschwelle `gs` ist hier sehr niedrig angesetzt, um einen Fall zu simulieren, bei dem die Fahnenbegehung mit einem anderen Probandenkollektiv durchgeführt worden ist als die Emissionsmessung. Der Unterschied in der mittleren Geruchsempfindlichkeit der Kollektive ist hier durch den Wert von `gs` wiedergegeben. Alternativ hätte man auch die Quellstärken `eq` umrechnen können.

<sup>5</sup>Bei *IBJodor* sind die beiden Quellen zu einer Quelle zusammengefaßt.



Startet man das Programm *IBJXodor* mit dem Aufruf

```
IBJXodor x/fahne
```

dann wird im Verzeichnis *x/fahne* folgender Protokoll-File erzeugt:

```
Odor Dispersion Model MEP0D, IBJXodor Version 1.8.d
Copyright Ing.-Büro Janicke, Dunum, Germany, 1999-2000

Licensed to: Ingenieurbüro Janicke, Dunum

STARTING at Sun Sep 10 11:45:41 2000
>F:\ibj\mepod\IBJXodor-1.8\wIBJXodor.exe
>F:\ibj\mepod\x\fahne\
## 24.08.92 10:00
Werte an den Aufpunkten:
  i:  Xp/(m)  Yp/(m)  Cnc/(GE/m3)      Fro      Frh
  1: 1079.0  922.0  2.168e-001  1.277e-001  1.000e+000
  2: 1004.0  916.0  4.673e-001  2.636e-001  1.000e+000
  3:  929.0  911.0  8.745e-001  4.547e-001  1.000e+000
  4:  854.0  906.0  1.376e+000  6.488e-001  1.000e+000
  5:  779.0  901.0  1.774e+000  7.678e-001  1.000e+000
  6:  705.0  895.0  1.822e+000  7.542e-001  1.000e+000
  7:  630.0  890.0  1.431e+000  6.000e-001  1.000e+000
  8:  555.0  885.0  8.172e-001  3.641e-001  1.000e+000
  9:  480.0  880.0  3.179e-001  1.530e-001  1.000e+000
 10:  405.0  875.0  7.581e-002  3.906e-002  0.000e+000
 11:  331.0  869.0  9.336e-003  5.005e-003  0.000e+000
 12:  256.0  864.0  3.128e-004  1.370e-005  0.000e+000

FINISHED at Sun Sep 10 11:45:42 2000
```

Im Protokoll-File meldet sich das Programm zuerst mit Namen und Versionsnummer. Es gibt den Zeitpunkt an, zu dem es compiliert wurde, und wann dieser Rechenlauf startet. Anschließend werden zur Kontrolle die Aufrufparameter aufgelistet, die über die Aufgabe „\*\*“ gesetzte Markierung und die Liste der Werte an den Aufpunkten. An den Punkten, wo die Geruchshäufigkeit *Fro* den Wert 0.1 überschreitet, ist die Geruchsstundenhäufigkeit *Frh* mit dem Wert 1 ausgewiesen, sonst ist sie null.

### 4.3.2 Wetterstatistik

In diesem Beispiel wird für eine 10 m hohe Punktquelle ohne Überhöhung die jährliche Geruchsbelastung über eine Wetterstatistik bestimmt.

Die Emission beträgt 10 MGE/h, die Aufpunkte liegen auf einem 50 m-Raster und überdecken ein Gebiet von  $500 \times 500 \text{ m}^2$ . Im Arbeitsverzeichnis steht die Wetterstatistik *anonym.dat* und folgender Eingabe-File *odrcmd.txt* :<sup>6</sup>

<sup>6</sup>Für *IBJodor* muß die Wetterstatistik per Hand in eine Folge von Einzelsituationen zerlegt und in den Eingabe-File eingefügt werden.



```
*d          'DIMENSIONIERUNG VON FELDERN UND RECHENGEBIET
*g          'GITTER VON AUFFUNKTEN
dr          50.0      'Maschenweite (m)
nx          11       'Anzahl der Punkte in X-Richtung
ny          11       'Anzahl der Punkte in Y-Richtung
xr          -250.0   'X-Koordinate des westlichen Randes (m)
yr          -250.0   'Y-Koordinate des südlichen Randes (m)
*yq        'QUELLEN: LAGE, AUSDEHNUNG UND QUELLSTÄRKEN
xq          0        'X-Koordinate der Quelle/Eckpunkt (m)
yq          0        'Y-Koordinate der Quelle/Eckpunkt (m)
hq          10.0     'Höhe der Quelle/Unterkante (m)
eq          2778.0   'Quellstärke (GE/s)
*p          ' [RUN] PARAMETER DER AUSBREITUNGSRECHNUNG
ha          10.0     'Höhe des Anemometers über Grund (m)
tm          3600     'Mittelungszeit für die Fahne (s)
z0          0.3200   'Rauigkeitslänge (m)
*w          "~anonym.dat"  '<FILENAME> WETTERSTATISTIK ABARBEITEN
*s          "cnc+fro+frh"  '[CNC|FRH|FRO] ERGEBNISSE AUSSCHREIBEN
fa          1.0E+03   'Skalierungsfaktor
fo          "%5.1f"   'Darstellungsformat
se          "j-.i+"   'Indexfolge
*e          'ENDE DER PARAMETERDATEI
```

Das Programm *IBJXodor* wird mit dem Aufruf

```
IBJXodor x/statistik
```

gestartet. Nach einer Rechenzeit von etwa 1 Minute stehen im Arbeitsverzeichnis neben dem Protokoll-File `odrlog.txt` auch die Ergebnis-Files `cnc.dmna` mit der mittleren Konzentration, `frh.dmna` mit der Häufigkeit der Geruchsstunden und `fro.dmna` mit der kumulativen Geruchshäufigkeit. Die Ergebnis-Files sind Text-Files, die mit einem normalen Editor angesehen und ausgedruckt werden können. Sie können auch per *drag and drop* aus dem Explorer (MS Windows) in ein Arbeitsblatt von Excel kopiert werden. Im folgenden ist `frh.dmna` aufgelistet:

```
mode "text"
sequ "j-.i+"
form "%5.1f"
fact 1.000e+003
xmin -250
ymin -250
delt 50
vldf P
mark "Häufigkeit von Geruchsstunden"
title "TEST"
ilbl -250 -200 -150 -100 -50 0 50 100 150 200 250
jlbl -250 -200 -150 -100 -50 0 50 100 150 200 250
dims 2
size 4
lowb 0 0
ghhb 10 10
*
13.2 13.8 16.8 21.7 29.0 34.7 33.8 28.1 22.5 16.8 14.4
15.5 16.9 19.6 25.5 37.8 51.3 48.8 34.8 26.2 20.8 16.3
19.6 22.6 25.6 31.5 47.5 75.6 72.7 55.0 38.3 25.3 20.9
21.5 27.6 35.7 40.9 54.9 106.5 119.0 91.4 56.3 31.2 23.4
19.7 28.9 41.5 57.7 54.7 104.2 165.0 135.7 71.1 40.5 24.2
16.9 26.3 41.0 65.3 68.5 0.0 124.6 116.0 69.0 38.0 22.5
16.7 25.3 40.0 67.6 81.0 66.8 70.2 70.4 45.2 28.3 18.6
18.5 25.7 40.2 57.7 68.7 67.5 53.6 43.0 31.0 19.4 15.0
19.0 24.4 33.6 43.0 48.3 48.4 37.0 27.6 20.8 15.0 12.5
17.0 21.2 25.8 30.2 32.9 32.2 25.3 18.3 14.4 12.1 9.7
16.1 17.7 20.9 22.5 22.8 21.4 18.1 14.4 11.9 9.5 8.5
***
```



Da die Werte mit dem Faktor 1000 multipliziert sind, stellen sie die Häufigkeit in Promille der Jahresstunden dar. Die nach dem „\*“ beginnende Tabelle ist so angeordnet, wie die Aufpunkte auf der Landkarte erscheinen würden: Der Index  $i$  ( $x$ -Richtung) läuft längs einer Zeile, der Index  $j$  ( $y$ -Richtung) läuft rückwärts von oben nach unten. Einzelheiten zu dem Aufbau eines solchen Files sind in Anhang A beschrieben.



## A Dateistruktur

Alle Dateien, die Ein- oder Ausgabefelder (Tabellen) repräsentieren, sind nach dem gleichen Prinzip aufgebaut. Sie enthalten zuerst einen Kopf, in dem alle Angaben zur Struktur und Darstellung der Tabelle stehen. Es folgt der Rumpf mit der eigentlichen Tabelle. Dieser Rumpf kann unmittelbar an den Kopf angehängt sein, wenn die Tabelle formatiert ausgeschrieben wird. Bei unformatierter Darstellung bildet der Rumpf immer einen eigenen File. In diesem Binär-File stehen die Tabellenelemente so, wie sie intern dargestellt sind, unmittelbar hintereinander ohne jegliche Steuerzeichen. Ein Tabellenelement kann eine einzelne Zahl (Datenelement) oder ein Verbund von mehreren Zahlen (*record*) sein.

Der Kopf ist ein Text-File mit der Namensweiterung „.dmna“, der aus einzelnen Zeilen besteht. In jeder Zeile ist ein Parameter definiert. Der Name des Parameters steht zu Beginn der Zeile, gefolgt von einem oder mehreren Werten. Zulässige Trennungszeichen sind Leerzeichen, Tabulator und Semikolon, die einzeln oder kombiniert verwendet werden können. Die Zeile kann durch LF oder CR+LF abgeschlossen sein.

Neben den vom Programm benötigten Parametern kann der Kopf auch weitere Parameter enthalten. Parameter, die das Programm nicht kennt, werden ignoriert. Der Kopf endet mit einer Zeile, die zu Anfang einen Stern, also \*, enthält. Mit der nächsten Zeile beginnt der Rumpf, sofern er mit dem Kopf zusammen einen einzigen File bildet. Die (formatierten) Tabellenelemente im Rumpf werden durch Leerzeichen, Semikolon, Tabulator, CR oder LF getrennt. Die Tabelle wird durch eine Zeile, die mit drei Sternen beginnt, beendet.

Folgende Parameter im Kopf der Datei werden vom Programm erkannt und interpretiert (die Namen müssen klein geschrieben und genau 4 Zeichen lang sein):

**buff** integer(1)

Anzahl der Bytes, die der Einlesepuffer für die Datenzeilen mindestens aufnehmen können muß.

**data** string(1)

Name des Files, der die eigentliche Tabelle enthält. Ist **data** nicht spezifiziert oder hat es den Wert „\*“, dann wird bei formatierter Ausgabe die Tabelle in den gleichen File geschrieben wie der Kopf. Bei unformatierter Ausgabe wird der File-Name des Kopfes übernommen, er erhält aber statt „.dmna“ die Endung „.dmnb“. Falls in **data** eine Pfadangabe gemacht ist, gilt diese relativ zu dem Verzeichnis, in dem der Kopf gespeichert ist.

**dims** integer(1)

Anzahl der Dimensionen (maximal 5).

**fact** float(1)

Faktor, mit dem bei formatierter Ausgabe alle Datenelemente vom Typ *float* oder *double* multipliziert werden, bevor sie im angegebenen Format ausgeschrieben werden. Bei der Eingabe formatierter Daten werden diese Datenelemente nach dem Einlesen durch **fact** dividiert.



**form** string(1)

Format, nach welchem bei formatierter Speicherung die Daten abgelegt sind. Bestehen die Tabellenelemente des gespeicherten Feldes aus mehreren Datenelementen, dann ist für jedes Datenelement eine Formatangabe erforderlich, und alle Einzelformate verkettet ergeben die Zeichenkette **form**.

$Format = Format_1 Format_2 \dots$

$Format_i = Name\%(*Factor) Length.PrecisionSpecifier$

Es bedeuten:

*Name* Name des Datenelementes (optional).

*Factor* Skalierungsfaktor (optional einschl. Klammern).

*Length* Länge des Datenfeldes.

*Precision* Anzahl der Nachkommastellen (bei *float*-Zahlen).

*Specifier* Umwandlungsangabe.

Der Skalierungsfaktor *Factor* wird genauso gehandhabt wie der Parameter **fact** und wirkt zusätzlich. Die Längenangabe *Length* ist die Mindestlänge des Datenfeldes. Sie kann überschritten werden, wenn dies zur korrekten Darstellung der Zahl erforderlich ist. Zwischen den Zahlen steht immer mindestens ein Trennungszeichen.

Folgende Umwandlungsangaben sind möglich:

| <i>Spec.</i> | Typ              | Länge | Beschreibung                  |
|--------------|------------------|-------|-------------------------------|
| c            | <i>character</i> | 1     | einzelne Buchstaben           |
| d            | <i>integer</i>   | 4     | Dezimalzahl                   |
| x            | <i>integer</i>   | 4     | Hexadezimalzahl               |
| f            | <i>float</i>     | 4     | Festkommazahl (ohne Exponent) |
| e            | <i>float</i>     | 4     | Gleitkommazahl (mit Exponent) |

Den Angaben **f** und **e** kann ein 1 vorangestellt sein (*double* mit Länge 8 Bytes), den Angaben **d** und **x** ein **h** (*short integer* mit der Länge 2 Bytes).

Gleichartige Formatangaben können zusammengefaßt werden:

$vx\%5.2fvz\%5.2fvz\%5.2f$  ist äquivalent zu  $vx\%[3]5.2f$

**hghb** integer(**dims**)

Höchster Indexwert für die verschiedenen Laufindizes.

**lowb** integer(**dims**)

Niedrigster Indexwert für die verschiedenen Laufindizes.

**mode** string(1)

Bei **binary** sind die Daten unformatiert gespeichert, sonst formatiert.

**sequ** string(1)

Angabe, in welcher Indexfolge die Daten gespeichert sind. Normalerweise läuft der am weitesten rechts stehende Index am schnellsten (C-Konvention). Dies entspricht bei einem 3-dim. Feld  $A_{ijk}$  der Angabe **i+:j+:k+**. FORTRAN speichert gemäß **k+:j+:i+**. Ein Minuszeichen statt des Pluszeichens bedeutet,



daß der betreffende Index rückwärts läuft. Es können auch Teilbereiche ausgewählt werden:  $j=10..1/1:i=5..25/1:k=1$ . Die Angabe  $/n$  bedeutet, daß der betreffende Index des Ausschnittes mit dem Wert  $n$  anfängt. Wird mit `sequ` ein Ausschnitt des Datenfeldes definiert, dann beziehen sich die Indexgrenzen `lowb` und `hghb` auf die ursprünglichen Indexdefinitionen.

`size integer(1)`

Länge der einzelnen Daten (*record size*) in Bytes. Bei formatierter Speicherung muß die aus der Formatangabe resultierende Summe der Längen der einzelnen Datenelemente gleich `size` sein.

### Beispiel:

Aus einem Feld von Gleitkommazahlen  $A_{ijk} = 100i + 10j + k$ ,  $i = 0..5$ ,  $j = 2..4$ ,  $k = 0..3$  wird ein Ausschnitt der horizontalen Schicht  $k = 1$  gespeichert:

```
form %4.1f
mode text
sequ j-:i=1..3/1:k=1
fact 1.000e-001
dims 3
size 4
lowb 1 2 1
hghb 3 4 1
*
14.1 24.1 34.1
13.1 23.1 33.1
12.1 22.1 32.1

***
```



## B Implementierte HPGL-Befehle

Ein HPGL-File ist ein Text-File mit Plot-Befehlen, die auf der Vorstellung eines Feder-Plotters beruhen. Ein Plot-Befehl besteht aus zwei Großbuchstaben, gefolgt von Argumenten. Argumente sind durch Leerzeichen oder Kommata separiert. Plot-Befehle können durch ein Semikolon beendet werden, andernfalls enden sie automatisch mit dem Beginn des nächsten Befehls. Einige Befehle, die Text als Argument erhalten, werden mit einem besonderen Zeichen (*terminator*) abgeschlossen. Dieses Zeichen kann vom Benutzer mit dem Befehl DT jederzeit frei definiert werden.

Primäre Maßeinheit in HPGL ist die *plotter unit* = 0.025 mm. Der Benutzer kann mit dem Befehl SC eigene Einheiten definieren (*user units*).

Es folgt eine alphabetische Auflistung der implementierten HPGL-Befehle mit einer Kurzbeschreibung:

DI *value<sub>x</sub>, value<sub>y</sub>; define inclination*

Festlegung der Schriftorientierung. Die beiden ganzzahligen Werte definieren die Steigung der Schriftgrundlinie (0 1: 90 Grad gegen den Uhrzeigersinn, 1 1: 45 Grad usw.).

CP*spaces, lines; character plot*

Die momentane Position der Feder wird um *spaces* Leerstellen nach rechts und um *lines* Zeilenabstände nach oben verschoben (mit erhobener Feder).

DT*Terminator; define terminator*

Es wird das Zeichen definiert, mit dem das Ende eines Textes angezeigt wird. Standardmäßig ist dies das ANSI-Steuerzeichen ETX (*end of text*, kodiert als 0x03). In den Beispielen wird hier immer das Zeichen @ verwendet.

EP; *edge polygon*

Der Polygonzug im Polygon-Buffer wird mit der aktuellen Feder und dem aktuellen Linientyp gezeichnet.

ER *dx,dy; edge rectangle relative*

Es wird ein Rechteck mit den Kantenlängen *dx* und *dy* gezeichnet. Startpunkt ist die linke untere Ecke auf der momentanen Zeichenposition.

FP; *fill polygon*

Das im Polygon-Buffer definierte Polygon wird als Fläche gezeichnet und dabei mit dem aktuellen Füllraster ausgefüllt.

FT *Type,Option1,Option2; fill type*

Füllraster festlegen; bei *Type*=1 wird mit der aktuellen Feder dicht ausgefüllt. Bei *Type*=11 wird mit *Option1* die Nummer des Bitrasters (> 0) und mit *Option2* die zu verwendende Feder (Farbnummer) ausgewählt.

IN; *initialize*

Das Grafik-System wird initialisiert und es werden eine Reihe von Parametern auf Standardwerte gesetzt.

IP  $P_{1x}, P_{1y}, P_{2x}, P_{2y}$ ; *input  $P_1$  and  $P_2$* 

Das durch  $P_1$  (linke untere Ecke) und  $P_2$  (rechte obere Ecke) definierte Fenster kann mit SC skaliert und zum Zeichnen verwendet werden. Koordinatenangaben sind in *plotter units*.

IW  $P_{1x}, P_{1y}, P_{2x}, P_{2y}$ ; *input window*

Das durch  $P_1$  (linke untere Ecke) und  $P_2$  (rechte obere Ecke) in *user units* definierte Fenster wird als beschreibbares Zeichenfeld definiert (*clipping area*). Wird IW ohne Parameter aufgerufen, werden die *clipping*-Grenzen entfernt.

LB *Text Terminator label*

Der Text *Text* wird an der momentanen Zeichenposition ausgeschrieben (mit der aktuellen Schrift). Der Text muß mit dem durch DT festgelegten *Terminator* beendet sein.

LO *Origin; label origin*

Für das Schreiben von Texten wird der Referenzpunkt festgelegt (1: links unten, 2: links Mitte, ..., 9: rechts oben). Erhöhung der Werte um 10 (Werte 11 bis 19) bewirken, daß zwischen Referenzpunkt und Text ein kleiner Abstand gelassen wird.

LT [ $n$ ]; *line type*

Es wird der Linientyp festgelegt. Ist kein Argument angegeben, wird eine durchgezogene Linie gezeichnet, sonst eine unterbrochene ( $n = 0$  : nur Anfangspunkt;  $n = 1$  : punktiert;  $n = 2$  : gestrichelt mit großem Zwischenraum;  $n = 3$  : gestrichelt mit kleinem Zwischenraum;  $n = 4$  : strichpunktiert;  $n = 5..8$  : Kombinationen aus langen und kurzen Strichen und Punkten).

PA  $x_1, y_1, \dots, x_n, y_n$ ; *pen absolute*

Es werden von der momentanen Position aus die Punkte  $(x_1, y_1)$  bis  $(x_n, y_n)$  angefahren. Wenn als letztes PD aktiv war, wird dabei eine Linie gezogen.

PC  $pen, red, green, blue$ ; *pen color*

Für die Feder *pen* wird die Farbe mit den Farbanteilen *red*, *green* und *blue* festgelegt. Die Werte der Farbanteile liegen im Bereich 0 bis 255. Höhere Farbanteile bedeuten hellere Farben (additive Mischung).

PD  $x_1, y_1, \dots, x_n, y_n$ ; *pen down*

Linie von der momentanen Position aus durch die Punkte  $(x_1, y_1)$  bis  $(x_n, y_n)$  ziehen. Wenn zuletzt PR aktiv war, werden die Koordinaten jeweils als relative Verschiebungen zur aktuellen Position interpretiert.

PM  $m$ ; *polygon mode*

Definition eines Polygonzuges. Bei  $m = 0$  wird der *polygon mode* gestartet und es werden die aktuelle Zeichenposition und alle weiteren, durch PU oder



PD definierten Punkte in den Polygon-Buffer geschrieben. Bei  $m = 1$  wird der momentane Polygonzug abgeschlossen, so daß mit dem nächsten Punkt ein neuer Polygonzug begonnen wird. Bei  $m = 2$  wird der *polygon mode* beendet, und der eingegebene Polygonzug steht im Polygon-Buffer. Er kann anschließend mit EP als Linie oder mit FP als Fläche gezeichnet werden.

PR  $x_1, y_1, \dots, x_n, y_n$ ; *pen relative*

Von der momentanen Position aus die Verschiebungen  $(x_1, y_1)$  bis  $(x_n, y_n)$  durchführen. Wenn zuletzt PD aktiv war, wird dabei eine Linie gezogen.

PS *length, width*; *plot size*

Die gesamte Bildgröße wird auf eine Breite *length* und eine Höhe *width* festgelegt. Die Werte sind in *plotter units* anzugeben. Dieser Befehl muß unmittelbar nach IN erscheinen.

PU  $x, y$ ; *pen up*

Als neue Zeichenposition den Punkt  $(x, y)$  wählen.

PW  $w$ [*pen*]; *pen width*

Linienbreite auf  $w$  mm setzen. Ist *pen* angegeben, gilt dies nur für die Feder *pen*, sonst für alle.

RR  $dx, dy$ ; *fill rectangle relative*

Rechteck mit den Kantenlängen  $dx$  und  $dy$  mit dem zur Zeit gültigen Füllmuster ausfüllen, Startpunkt ist die linke untere Ecke auf der momentanen Zeichenposition.

SC  $Xmin, Xmax, Ymin, Ymax$ ; *scale*

Das durch IP definierte Fenster wird skaliert, so daß der Punkt  $P_1$  dem Wertepaar  $(Xmin, Ymin)$  und der Punkt  $P_2$  dem Wertepaar  $(Xmax, Ymax)$  entspricht (Festlegung von *user units*).

SI *width, height*; *absolute character size*

Legt die Breite und Höhe von Buchstaben fest, Maßeinheit sind cm.

SP *pen*; *select pen*

Die Feder mit der Nummer *pen* auswählen.