

Technische Anleitung „Effektive Maschenweite“

- **Zerschneidungsgeometrie:
Arbeitsschritte in *ArcView/ArcInfo***
- **ArcView- extension: „meff_tool“**
- **Teilraummethoden: kurze Erläuterung**

Hans-Georg Schwarz-v. Raumer und Heide Esswein

2006

Kontakt:

Hans-Georg Schwarz-v. Raumer
Institut für Landschaftsplanung und
Ökologie Universität Stuttgart
Keplerstraße 11
70174 Stuttgart
svr@ilpoe.uni-stuttgart.de
Tel.: 0711/68584145

Heide Esswein
Institut für
Landschaftsplanung und
Ökologie Universität Stuttgart
Keplerstraße 11
70174 Stuttgart
he@ilpoe.uni-stuttgart.de
Tel.: 0711/68584119

I Zerschneidungsgeometrie: Arbeitsschritte in ArcView/ArcInfo

Die Bearbeitung der ATKIS-Daten in ArcInfo erfordert folgende Schritte:

1. Alle benötigten *shapefiles* werden in *ArcInfo* zu *coverages* umgewandelt.

Befehl: shapearc <in shapefile> <out cover>

Anmerkung: Die Umwandlung kann auch in ArcGIS durchgeführt werden, es ist jedoch darauf zu achten, dass die Attribute der *shapefiles* nicht mitgeführt werden, da sonst der anschließende *append*-Befehl nicht funktioniert.

Für Polygondateien muss mit dem Befehl „*regionpoly*“ eine subclass für die Attribute angegeben werden. Hierdurch werden die Attribute erhalten und das *coverage* besitzt gleichviel Polygone wie das *shapefile*.

2. Mit dem Befehl „*append*“ werden alle *coverages*¹ zusammengeführt. Anschließend folgt der „*clean*“ Befehl, damit die Linienstruktur in Polygone umgewandelt wird. Das Ergebnis ist ein Polygoncoverage, das im Folgenden mit den weiteren Dateien polygonaler Struktur vereinigt werden kann.

append outcover

1st cover

2nd cover

3rd

.

.

8th end

clean appendcover outcover 10 1.01 poly

3. Die entstehende Netzdatei wird nun mit den weiteren Polygondateien verschnitten. Hierzu wird in die Attributtabelle der Polygondateien jeweils eine neue Spalte angefügt. Die Spalte erhält den Wert 1 für das jeweilige Thema, damit diese Flächen später identifiziert und wieder von der Datei der unzerschnittenen Räume abgezogen werden können. Dieser Schritt wurde in *ArcView* durchgeführt. Danach werden in *ArcInfo* die Polygondateien mit *union* mit der Netzgeometrie vereinigt.

¹ Ursprünglich werden damit lediglich die Liniendateien zusammengeführt. Da jedoch bei Verwendung von Verkehrszählungsdaten innerhalb der Ortslagen häufig fehlende Straßenabschnitte zu Lücken in der Liniengeometrie führen, werden die Polygone der Ortslagen ebenfalls schon in diesem Schritt zusammengeführt. Da jedoch die Information über die Attribute dabei verloren geht, müssen die Ortslagen dennoch mit *union* ein zweites Mal verknüpft werden, um sie anschließend von den unzerschnittenen Räumen zu unterscheiden.

Befehl: *union Netzgeometrie unionGeometrie outcover 1.101 join*

Mit diesem Befehl werden nun nacheinander alle Polygondateien zur Netzgeometrie dazugefügt.

- Die weitere Verarbeitung der Netzgeometrie erfolgt nun in ArcView. Aus **der Netzgeometrie** werden mit „*query*“ alle Flächen, die nicht als unzerschnittene Flächen betrachtet werden sollen, sondern als Barriere (z.B: Ortslagen) selektiert, dann wird die Selektion umgekehrt (*change selection*), womit alle unzerschnittenen Räume selektiert sind. Diese werden mit dem Befehl „*convert to shapefile*“ in ein neues *shapefile* überführt. Man erhält die Datei, die das Ergebnis für die unzerschnittenen Räume enthält.

I.1 Berechnung der effektiven Maschenweite in Excel

Die mit der Zerschneidung in *ArcInfo* und *ArcView* gewonnenen Daten können nun im Programm *Excel* mit der Formel der effektiven

Maschenweite:

$$m_{\text{eff}} = F_g \cdot \sum_{i=1}^n \left(\frac{F_i}{F_g} \right)^2$$

weiterverarbeitet werden. Der Datenaustausch erfolgt über eine *dBase*-Tabelle. Man bildet zunächst die Quadratesumme aller Flächeninhalte und teilt diese durch die Gesamtfläche des jeweils betrachteten Raumes. Hier ist zu beachten, dass der gesamte Flächeninhalt des Bezugsraumes in den Nenner gesetzt wird und nicht die Summe aller verbleibenden (zerschnittenen und unzerschnittenen) Flächen. (Eine Berechnung mit der Summe der Flächen wäre auch möglich, führt aber zu einer anderen Aussage; vgl. Jaeger 2002: 146ff.)

II ArcView-Erweiterung: „meff-tool“

Der Wert der effektiven Maschenweite kann auch in ArcView mit der extension „meff.avx“ berechnet werden. Mit Hilfe dieses GIS-Werkzeugs wird auch die Analyse der Zerschneidung von Teilräumen unterstützt. Das Werkzeug besteht nach Aktivierung aus einem Menübutton ‚Meff‘, welcher auf der Benutzeroberfläche ergänzt wird. Über dieses Menü ist dann neben der Durchführung einer Gesamtraumanalyse eine Teilraumanalyse möglich. Die extension ist nur englischsprachig verfügbar.

II.1 Benutzerführung Gesamtraumanalyse/Anwendungsbeispiel

Mit dem Menü-Item „complete analysis“ im Menü „Meff“ (Abb. 1) wird ein Benutzerdialog gestartet, der zunächst auffordert, aus der Themenliste die Zerschneidungsgeometrie zu bezeichnen (Abb. 2).

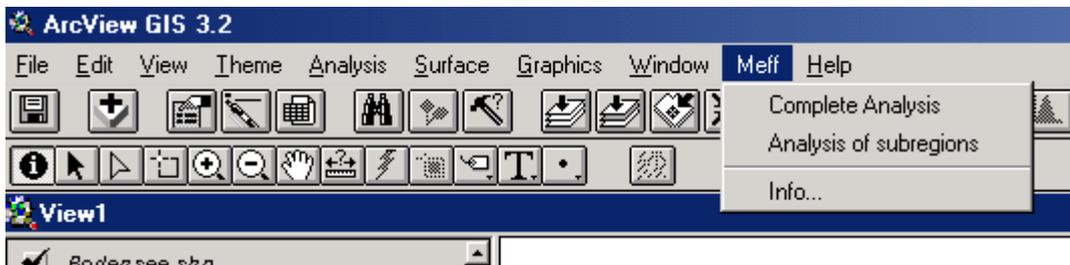


Abb. 1: Menü 'Meff' in meff.apr

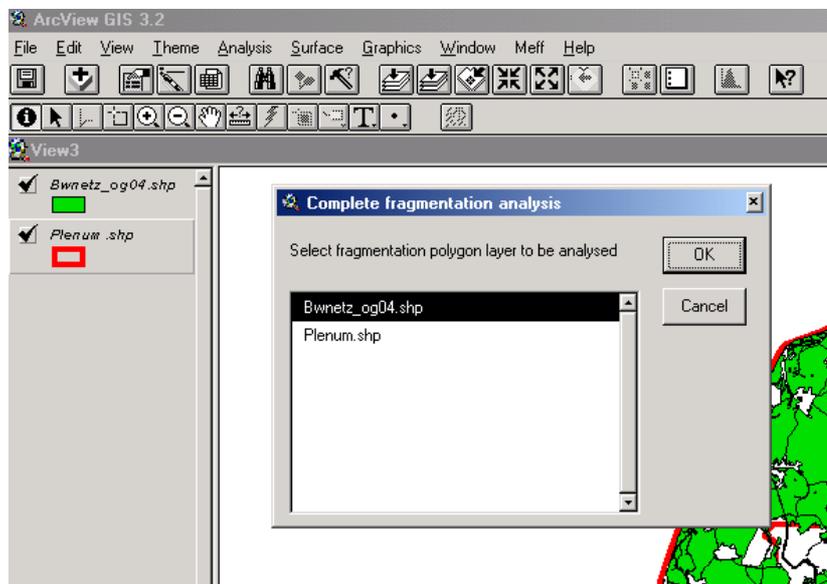


Abb. 2: Auswahl der Zerschneidungsgeometrie

Ist diese Auswahl erfolgt, startet der Rechenvorgang. Bevor dieser beendet wird, erfolgt eine Abfrage nach der Bezugsregion bzw. der Gesamtfläche, welche betrachtet wird (Abb. 3).

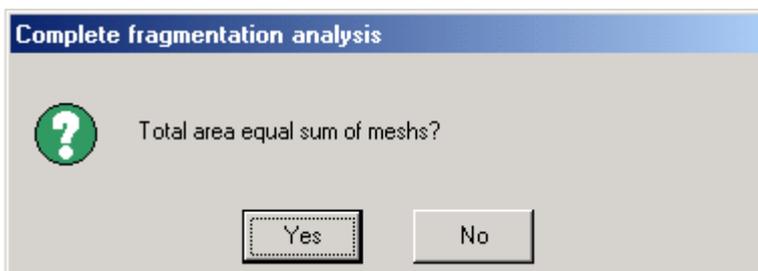


Abb. 3: Frage nach Bezugsfläche

Hierzu muss erläutert werden, dass innerhalb des Landschaftszerschneidungsprojektes bisher immer die Formel der effektiven Maschenweite so angewendet wurde, dass die Bezugsfläche, also der Wert, der im Nenner steht, der betrachteten Gesamtfläche (im Falle BW der

Landesfläche) entspricht. Generell kann die Formel jedoch auch anders interpretiert werden und es kann die Summe der Fläche aller unzerschnittenen Räume in den Nenner gesetzt werden. Diese Option kann hier getroffen werden. Wenn die Gesamtfläche der Flächensumme der UZR entsprechen soll, wählt man „ja“. Wird nein gewählt so kann man die entsprechende Fläche manuell eingeben (Abb. 4).

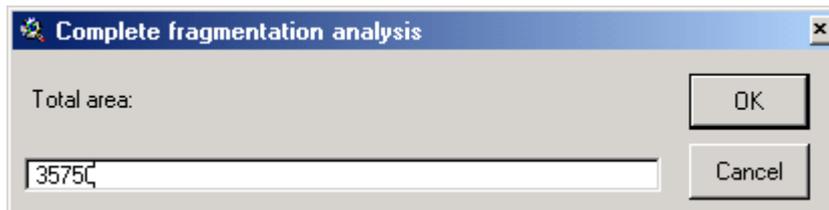


Abb. 4: Eingabefenster für Bezugsfläche (in km²)

Der Rechenvorgang wird dann zu Ende geführt und in einem Ergebnisfenster (Abb. 5) werden neben der effektiven Maschenweite auch die Flächeninhalte des kleinsten und des größten Polygons sowie der arithmetische Mittelwert der Flächeninhalte aller Polygone ausgegeben. Zusätzlich wird nochmals die Fläche angezeigt, die für die Berechnung in den Nenner eingesetzt wurde.

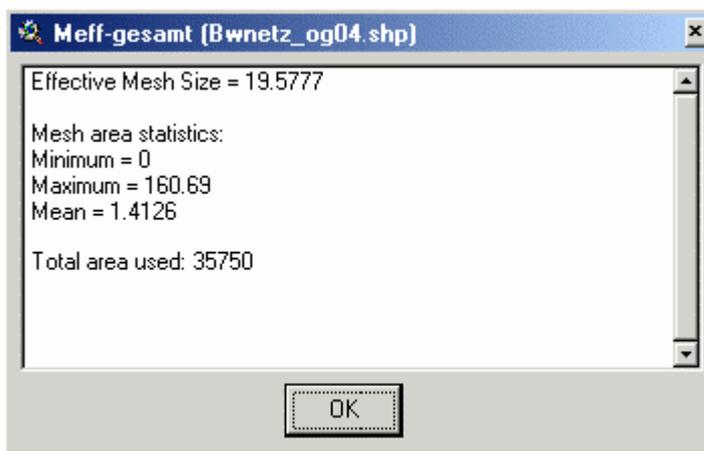


Abb. 5: Ergebnis
Gesamtraumanalyse

II.2 Benutzerführung Teilraumanalyse / Anwendungsbeispiel

Für eine Analyse von Teilräumen hinsichtlich deren Zerschneidungsgrad ist es notwendig mindestens zwei Themen einem View hinzugefügt zu haben: Ein Thema das die Zerschneidungsgeometrie darstellt (und den Gesamttraum überdeckt) sowie ein Thema das die zu analysierenden Teilräume beinhaltet (s. Abb. 6).

Mit dem Menü-Item „Analysis of subregions“ im Menü „Meff“ (Abb. 7) wird ein Benutzerdialog gestartet, der zunächst auffordert, aus der Themenliste die Zerschneidungsgeometrie zu bezeichnen. In einem nächsten Schritt wird das zu analysierende Teilraumthema abgefragt (Abb. 8).

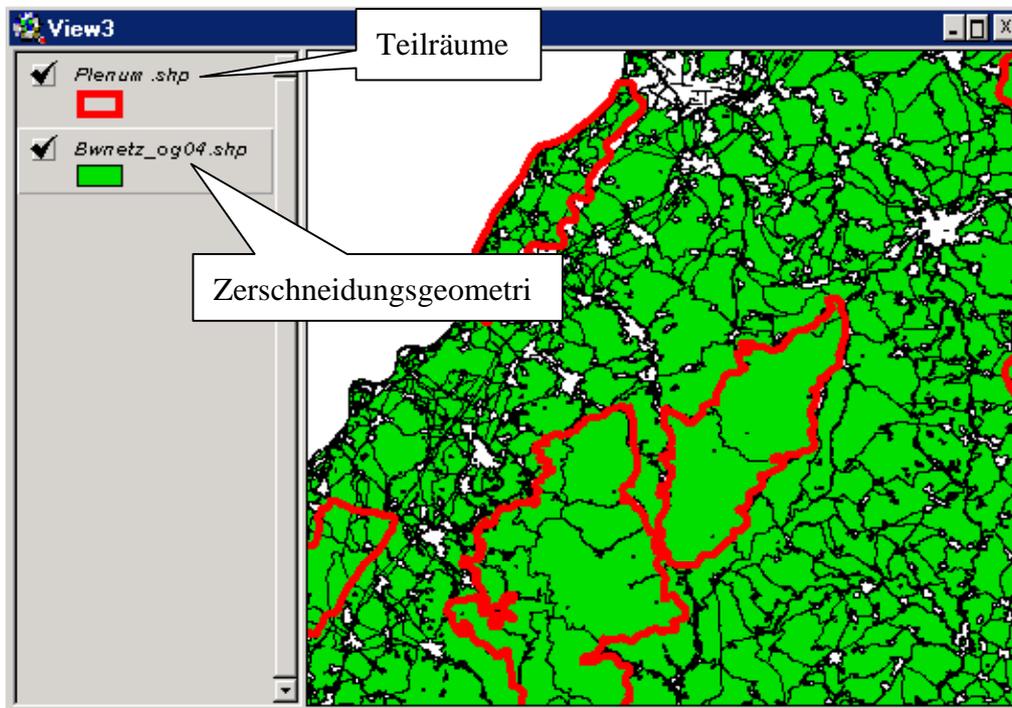


Abb. 6: Layers bei einer Teilraumanalyse

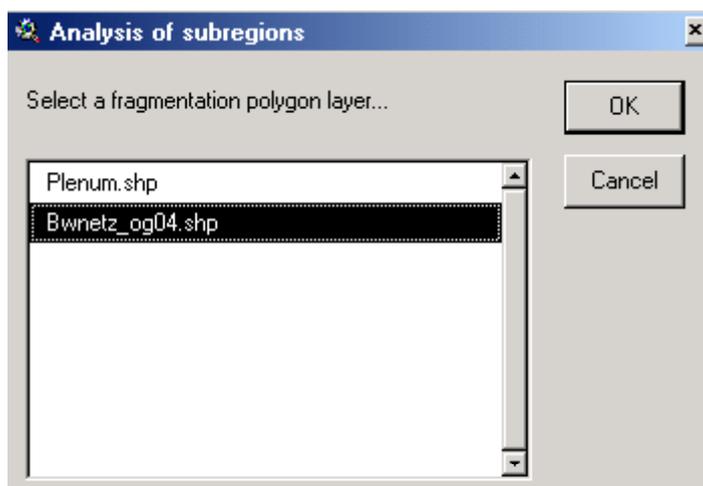


Abb. 7: Auswahldialog für Zerschneidungsgeometrie

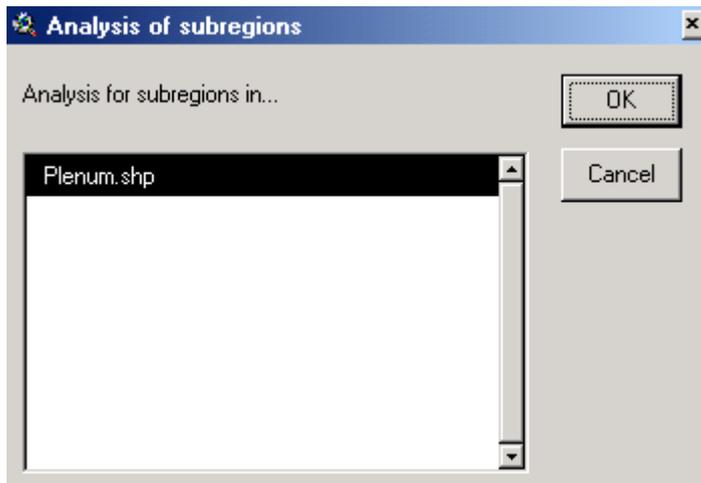


Abb. 8: Auswahldialog für Teilraumgeometrie

Als dritter Schritt ist die Analysemethode zu spezifizieren. Sechs Methoden stehen hier zur Auswahl (Abb. 9): dem Teilraum zugeordnet und in die Berechnung einbezogen werden aus der Zerschneidungsgeometrie

- (1) alle Polygone, die eine gemeinsame Schnittfläche mit dem Teilraum besitzen,
- (2) alle Polygone, die komplett von dem Teilraum umschlossen werden,
- (3) nur jene Polygone, deren Zentrum (Zentroid) im Teilraum liegt.
- (4) Bei der vierten Methode werden die Teilraumgrenzen als zerschneidende Linien akzeptiert, d.h. es wird zunächst eine gemeinsame Geometrie (Verschneidungsgeometrie) aus Zerschneidungspolygonen und Teilraumpolygonen gebildet. Dem Teilraum werden dann wiederum all jene Polygone aus der Verschneidungsgeometrie für die Berechnung zugeordnet, die er vollständig überdeckt.
- (5) Alle Polygone die vom Teilraum „angeschnitten“ werden, werden anteilig in die Berechnung einbezogen,
- (6) Wie (5) jedoch werden die Anteile doppelt gerechnet.

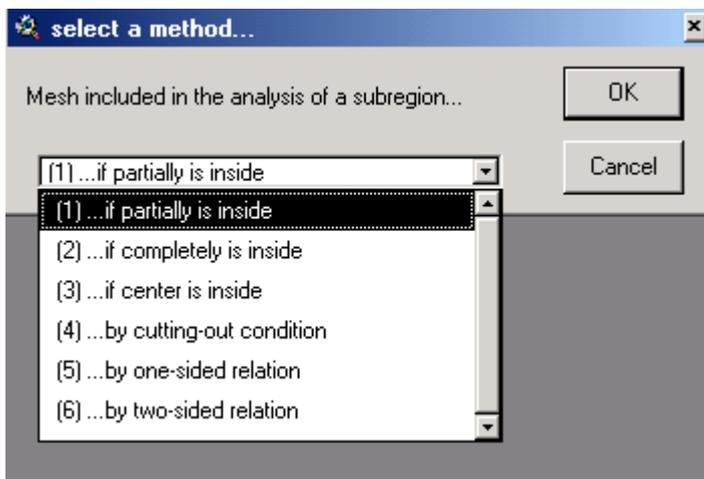


Abb. 9: Auswahl der Analysemethode

Im vierten Schritt der Benutzerführung für die meff-Teilraumanalyse wird danach gefragt, ob eine Flächenstatistik als Ausgabe gewünscht wird (s.u.). Nach der Beantwortung dieser Frage wird die Teilraumanalyse durchgeführt (Abb. 10).

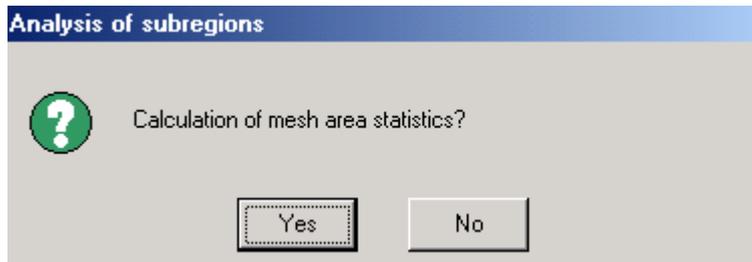


Abb. 10: Abfrage nach Ergebnisausgabe

Ergebnisausgabe. Das Ergebnis der Berechnungen wird in zusätzlichen Spalten (Felder) in der Attributstabelle für die Teilraumpolygone festgehalten (s. Abb. 11). Dabei wird auf jeden Fall der Tabelle eine Spalte hinzugefügt, in welcher die ermittelte effektive Maschenweite (Kennwort ‚Meff‘) des Teilraums eingetragen ist. Die effektive Maschenweite ist dabei in km² dimensioniert, wobei davon ausgegangen wird, dass die Koordinaten der Ausgangsgeometrien in Metern (m) skaliert sind (d.h. beispielsweise im Gauß-Krüger-Koordinatensystem erfasst wurden).

Wurde im vierten Schritt der Spezifikationen für die Berechnung angegeben, dass eine Flächenstatistik erstellt werden soll, dann werden die Teilräume mit den folgenden Größen aus der Flächenstatistik charakterisiert: Flächeninhalt der kleinsten dem Teilraum zugeordneten Fläche (Kennwort ‚MinFi‘), Flächeninhalt der größten dem Teilraum zugeordneten Fläche (Kennwort ‚MaxFi‘) und arithmetisches Mittel aller dem Teilraum zugeordneten Flächen (Kennwort ‚MiwFi‘).

Die Feldnamen der hinzugefügten Spalten ergeben sich aus der wie folgt eingeführten Namenskonvention:

<Kennwort>_<Zerschneidungsgeometrie>_<Methode>,

wobei als <Methode> die Kennung (1), (2), (3), (4), (5) oder (6) entsprechend Abb. 6 verwendet wird.

Shape	Id	Meff_Unzersch_ATKIS.shp_(3)	MiwFi_Unzersch_ATKIS.shp_(3)	MinFi_Unzersch_ATKIS.shp_(3)	MaxFi_Unzersch_ATKIS.shp_(3)
Polygon	1	5.57	1.50	0.00	12.00
Polygon	2	13.83	3.26	0.00	18.52
Polygon	3	15.62	3.61	0.00	25.95
Polygon	4	6.79	0.88	0.00	15.24
Polygon	5	7.22	2.05	0.00	18.07
Polygon	6	2.87	0.84	0.00	8.05
Polygon	7	132.74	8.34	0.00	118.62

Abb. 11: Ergebnis einer Teilraumanalyse

Technischer Hintergrund. Das Werkzeug besteht aus vier in AVENUE programmierten Subroutinen : meff_exec, clip, meff_ganz und meff_teil (s.Anhang).

Die Subroutine **meff_exec** führt die eigentliche Berechnung der effektiven Maschenweite durch. Als Übergabeparameter benötigt die Routine drei Angaben vom aufrufenden Unterprogramm: Den Dateinamen der Zerschneidungsgeometrie, eine Identifikationsnummer und den Flächenumfang der als Referenzgesamtfläche bei der meff-Berechnung verwendet werden soll. Über alle Polygone der Zerschneidungsgeometrie wird, entsprechend der Formel für die effektive Maschenweite, die Summe der Quadrate der Polygonflächeninhalte gebildet und entweder zur Summe der Polygonflächeninhalte selbst oder aber zur als Parameter übergebenen Referenzfläche ins Verhältnis gesetzt (falls nicht als 0 übergeben). Die berechnete effektive Maschenweite wird an die aufrufende Routine nebst den Flächeninhalten des kleinsten und des größten Polygons sowie dem Mittelwert der Polygonflächeninhalte übergeben. Das Ergebnis wird aber auch mit der oben erwähnten Identifikationsnummer gekennzeichnet in eine Textausgabedatei ‚sum.txt‘ geschrieben. Diese Datei ist auch als ArcView-Tabelle lesbar.

Das Teilprogramm **clip** schließlich entstammt den System-Skripten welche ArcView offenlegt. Es dient generell dazu, in einer Geometrie nur jene Bereiche zu belassen, die innerhalb von Clip-Polygonen liegen. Im vorliegenden Anwendungsfall werden mit der Clip-Subroutine aus der Zerschneidungsgeometrie nur jene Teilbereiche ausgeschnitten, die mit den Polygonen der Teilräume überdeckt werden. Das clip-Systemskript wurde dahingehend überarbeitet, dass die Spezifikation der zu verarbeitenden Themen und der zu generierenden Shape-files nicht über Dialoge stattfindet, sondern mit den Übergabeparametern beim Skriptaufruf.

Meff_ganz führt eine Gesamtraumanalyse aus, wobei zunächst die oben erwähnten Dialoge bereitgestellt werden. Dann wird mit den getroffenen Spezifikationen **meff_exec** aufgerufen, und abschließend die von dort übergebenen Ergebnisse dargestellt.

Meff_teil führt eine Teilraumanalyse aus, wobei zunächst die oben erwähnten Dialoge bereitgestellt werden. Mit den gemachten Spezifikationen wird in einer Zählschleife über alle Polygone der Teilraumgeometrie jeweils ein der gewählten Analysemethode (s.o.) entsprechender Ausschnitt aus der Zerschneidungsgeometrie generiert. Die Auswahl für die Methoden (1)-(3), (5) und (6) wird dabei mit der Funktion‘Selectbythema gewonnen. Für Methode(4) hingegen wird das Skript **clip** ausgeführt. Der generierte Ausschnitt wird dann mit **meff_exec** bearbeitet, und abschließend das von dort übergebene Ergebnis in die Attributstabelle der Teilraumpolygone eingetragen.

III Teilraummethoden: kurze Erläuterungen

Zur Betrachtung des Zerschneidungsgrades einzelner Teilregionen wurden bisher zwei unterschiedliche Methoden verwendet. Das Ausschneideverfahren sowie das Mittelpunktverfahren ermöglichen eine Berechnung der Werte der effektiven Maschenweite für Teilräume. Die Ermittlung dieser Werte kann zum einen manuell über die Funktionen in ArcView geschehen („Geoprocessing Wizard“, „Select by Theme“). Einfacher, bzw. benutzerfreundlicher ist jedoch die Erweiterung für ArcView (meff-tool), die unter Punkt II beschrieben wird. Mit diesem „Werkzeug“ werden die Werte der effektiven Maschenweite berechnet und direkt in die Attributtabelle der Teilraumgeometrie eingetragen. Das Werkzeug enthält sechs unterschiedliche Methoden, nach denen die unzerschnittenen Flächen dem Teilraum zugeteilt werden. Methode (1) teilt alle unzerschnittenen Flächen, die zwischen den Teilräumen liegen, jeweils allen Teilräumen zu. Das heißt es findet keine eindeutige Zuweisung der Flächen statt. Dies führt in der Berechnung zu Inkonsistenzen, da die Maschenweite dann größer werden kann als die größte einbezogene Fläche. Methode (2) teilt die UZR, die zwischen den Teilräumen liegen keinem Teilraum zu, das heißt, die Flächen werden dem Teilraum nur dann zugeordnet wenn sie vollständig innerhalb des Teilraums liegen. Es verbleiben also UZR, die keinem Teilraum zugeordnet werden. Diese beiden Methoden wurden in unseren bisherigen Untersuchungen nicht angewendet. Verwendet werden seit Beginn der Berechnungen die Methoden (3) Mittelpunktverfahren (MpV) und (4) Ausschneideverfahren (AsV). Neu eingeführt wurden 2004 das (5) Einseitige Beziehungsverfahren (engl. one-sided relationship OsR) und das (6) zweiseitige Beziehungsverfahren (double-sided relationship). Zu diesen beiden Teilraumverfahren sind im Kasten die Formeln und Beispiele zur Erläuterung enthalten.

Kasten: Definition der effektiven Maschenweite innerhalb des Beziehungsverfahrens

Die Formel für m_{eff} lautet für das einseitige Beziehungsverfahren:

$$m_{\text{eff}}^{\text{FaV}} = \frac{1}{F_{\text{total}}} (F_1 \cdot F_1^{\text{erg.}} + F_2 \cdot F_2^{\text{erg.}} + F_3 \cdot F_3^{\text{erg.}} + \dots + F_n \cdot F_n^{\text{erg.}})$$

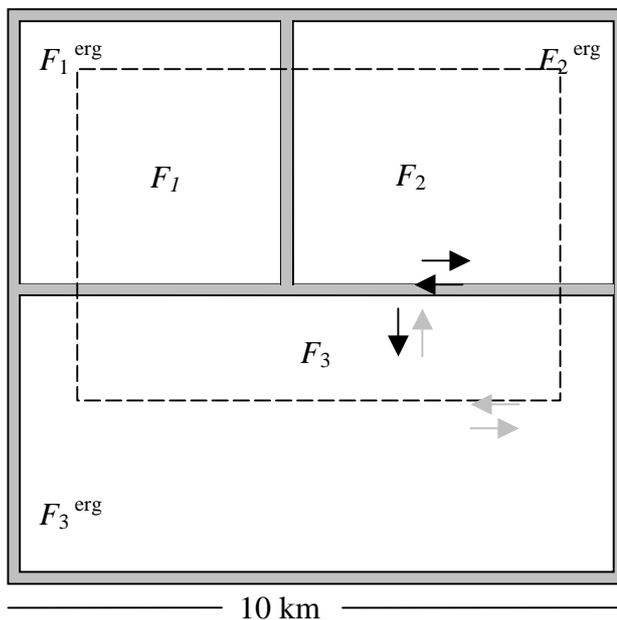
mit F_{total} = Gesamtfläche des Teilraumes

F_1, F_2 , usw. bis F_n = Größe von Fläche 1, 2, usw. bis n ;

n = Zahl der Flächen innerhalb des Teilraumes (ohne die Siedlungs- und Verkehrsflächen, denn diese zählen als Barrieren gegen die Fortbewegung von Tieren bzw. von Erholungssuchenden);

$F_1^{\text{erg.}}, F_2^{\text{erg.}}$, usw. bis $F_n^{\text{erg.}}$ = Größe von Fläche 1, 2, usw. bis n , wobei diese Flächen über die Teilraumgrenze hinaus fortgesetzt werden, bis sie von einer realen Barriere in der Landschaft begrenzt werden (im Fall, dass eine Fläche nicht mit der Teilraumgrenze in Berührung kommt, sondern vollständig innerhalb der Teilraumgrenzen liegt, ist $F_1^{\text{erg.}}$ gleich F_1 usw.).

Berechnungsbeispiel



Die Pfeile stellen eine erweiterte Interpretationsmöglichkeit dar. Die schwarz gefärbten Pfeile zeigen die Wege auf, die bei der Anwendung des einseitigen Beziehungsverfahrens für ein Tier oder einen Menschen in Frage kommen. Die grauen Pfeile zeigen alle sonst noch möglichen Wege auf.

$$F_1 = 3 \text{ km} \times 4 \text{ km} = 12 \text{ km}^2; \quad F_2 = 5 \text{ km} \times 4 \text{ km} = 20 \text{ km}^2; \quad F_3 = 8 \text{ km} \times 2 \text{ km} = 16 \text{ km}^2;$$

$$F_1^{\text{erg.}} = 4 \text{ km} \times 5 \text{ km} = 20 \text{ km}^2; \quad F_2^{\text{erg.}} = 6 \text{ km} \times 5 \text{ km} = 30 \text{ km}^2; \quad F_3^{\text{erg.}} = 10 \text{ km} \times 5 \text{ km} = 50 \text{ km}^2.$$

Damit wird die eff. Maschenweite berechnet zu:

$$m_{\text{eff}}^{\text{FaV}} = \frac{1}{F_{\text{total}}} (F_1 \cdot F_1^{\text{erg.}} + F_2 \cdot F_2^{\text{erg.}} + F_3 \cdot F_3^{\text{erg.}} + \dots + F_n \cdot F_n^{\text{erg.}})$$

$$= \frac{1}{48 \text{ km}^2} (12 \cdot 20 \text{ km}^4 + 20 \cdot 30 \text{ km}^4 + 16 \cdot 50 \text{ km}^4) = \frac{1640 \text{ km}^4}{48 \text{ km}^2} = 34,17 \text{ km}^2.$$

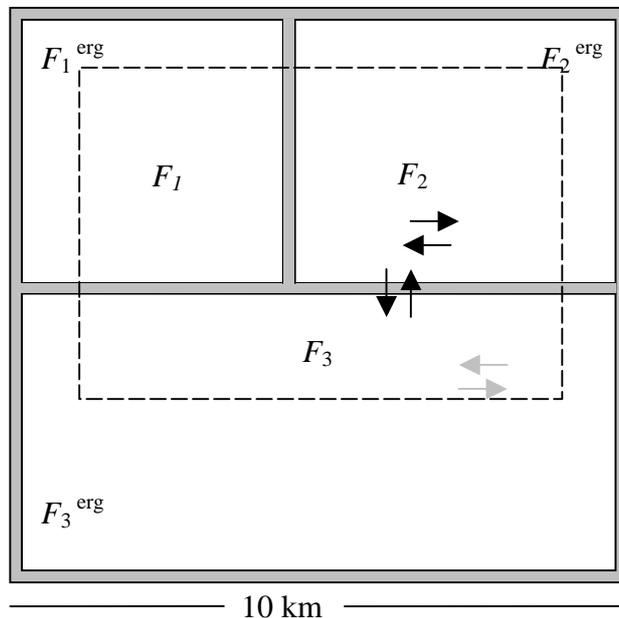
Dieser Wert ist größer als der Wert, der sich nach dem Ausschneideverfahren ergibt (16,67 km²).

Für das zweiseitige Beziehungsverfahren lautet die Formel:

$$m_{\text{eff}}^{\text{FaV}} = \frac{1}{F_{\text{total}}} (2F_1 F_1^{\text{erg}} - F_1^2 + 2F_2 F_2^{\text{erg}} - F_2^2 + 2F_3 F_3^{\text{erg}} - F_3^2 + \dots + 2F_n F_n^{\text{erg}} - F_n^2)$$

$$= \frac{1}{48 \text{ km}^2} (2 \cdot 12 \text{ km}^2 \cdot 20 \text{ km}^2 - 224 \text{ km}^2 + 2 \cdot 20 \text{ km}^2 \cdot 30 \text{ km}^2 - 400 \text{ km}^2 + 2 \cdot 16 \text{ km}^2 \cdot 50 \text{ km}^2 - 256 \text{ km}^2)$$

$$= \frac{2400 \text{ km}^4}{48 \text{ km}^2} = 50 \text{ km}^2$$



Im Vergleich zum einseitigen Beziehungsverfahren werden hier auch die Wege, die außerhalb des Teilraumes starten und in den Teilraum hineinführen miteinbezogen.

Im Folgenden werden die Vor- und Nachteile der verwendeten Methoden ausführlicher erläutert:

Zu (3) Mittelpunktverfahren:

Alle Flächen deren Mittelpunkt im betrachteten Teilraum liegen, werden in die Berechnung einbezogen. Dieses Verfahren eignet sich am besten für große Teilräume, für die man den natürlichen Zerschneidungsgrad berechnen möchte. Bei sehr kleinen Teilräumen, (z.B. Gemeinden) arbeitet dieses Verfahren zu ungenau, vor allem dann, wenn der Teilraum sehr große unzerschnittene Flächen berührt.

Zu (4) Ausschneideverfahren:

Die unzerschnittenen Flächen werden an der Teilraumgrenze „abgeschnitten“, das heißt, in die Berechnung fließen exakt die Flächenanteile ein, die innerhalb des Teilraumes liegen. Dieses Verfahren wird vor allem für die historische Analyse empfohlen, da es hierfür notwendig ist, dass die Grenzen zu allen Zeitpunkten stabil bleiben. Nachteil ist, dass künstliche Grenzen geschaffen werden. Die Werte der m_{eff} sind immer etwas niedriger als in der realen Situation.

Zu (5) Einseitiges Beziehungsverfahren:

In die Berechnung fließen alle Flächenanteile anteilig ein, die der Teilraum „anschneidet“. D.h. im Gegensatz zu den vorherigen Verfahren, bei denen die unzerschnittenen Flächen eindeutig zugeordnet werden, werden sie hier mehrfach zugeordnet. Eine große Fläche, die innerhalb dreier Teilräume liegt, wird **anteilig** jedem Teilraum zugerechnet. Ein einfaches Verständnisbeispiel ist im Kasten erläutert².

Zu (6) zweiseitiges Beziehungsverfahren:

Das Verfahren ist eine Erweiterung des einseitigen Beziehungsverfahrens. Im Gegensatz dazu werden die Wechselbeziehungen zwischen dem Teilraum und dem unzerschnittenen Raum jedoch erweitert um die Möglichkeit auch von „außen“ in die Teilfläche gelangen zu können. Es werden also zusätzlich zu den Wegen, die aus der Teilfläche in den angrenzenden Bereich des UZR führen auch die Wege einbezogen, die außerhalb der Teilfläche beginnen und in diese hineinführen. Deshalb eignet sich das Verfahren besonders gut für tierökologische Fragestellungen. Es ist jedoch aus mathematischen Gründen inkonsistent, wenn man es für Teilräume anwendet, die direkt aneinandergrenzen. Deshalb sollte man dieses Verfahren nur anwenden, wenn Teilräume vorliegen, die nicht direkt benachbart sind.

² Genaueres zu diesem Verfahren s. auch MOSER, B., JAEGER, J.A.G., TAPPEINER, U., TASSER, E., EISELT, B. (2007): Modification of the effective mesh size for measuring landscape fragmentation to solve the boundary problem. – Landscape Ecology, Im Druck, doi: 10.1007/s10980-006-9023-0.

Anhang

Subroutine meff_exec

```

theproj = av.GetProject
uFn = self.get(0) 'filename of shape that contains geometry to analyse
id = self.get(1) 'id tag (string)
Fg = self.get(2) 'complete area as number; if zero: Fg is set to sum_Fi

vt = ftab.make(srcname.make(uFn.asstring))

if (File.Exists(uFn))
then
  vt.seteditable(true)
  fi = vt.FindField( "Fi")
  fi2 = vt.FindField( "Fi2")
  g = vt.FindField( "g")

  if (fi <> nil) then
    if (MsgBox.YesNo ("Field Fi exists. Use?", "Fragmentation analysis", true) = false ) then
      calcstr = "[Shape].ReturnArea / 1000000"
      vt.Calculate( calcstr, fi)
    end
  end

  if (fi = nil)
  then fi = Field.Make("Fi", #FIELD_DOUBLE , 16, 8)
  vt.addfields({ fi})
  calcstr = "[Shape].ReturnArea / 1000000"
  vt.Calculate( calcstr, fi)
  end

  if (fi2 = nil)
  then fi2 = Field.Make("Fi2", #FIELD_DOUBLE , 16, 8)
  vt.addfields({ fi2})
  end

  if (g = nil)
  then g = Field.Make("g", #FIELD_CHAR , 6, 0)
  vt.addfields({ g})
  end

  calcstr = "[Fi]*[Fi]"
  vt.Calculate( calcstr, fi2)

  calcstr = id
  vt.Calculate( calcstr.quote, g)

  vt.seteditable(false)

  sumFile = filename.Make("$HOME"+"sum.txt")
  groupfield = vt.FindField( "g")
  statTab = vt.Summarize( sumFile, DText, groupfield, {fi,fi,fi,fi,fi2}, { #VTAB_SUMMARY_AVG,
#VTAB_SUMMARY_MIN, #VTAB_SUMMARY_MAX, #VTAB_SUMMARY_SUM, #VTAB_SUMMARY_SUM })
  areaavg = statTab.returnvaluenumber(statTab.findfield("Ave_fi"),0)
  areamin = statTab.returnvaluenumber(statTab.findfield("Min_fi"),0)
  areamax = statTab.returnvaluenumber(statTab.findfield("Max_fi"),0)
  sumFi2 = statTab.returnvaluenumber(statTab.findfield("Sum_Fi2"),0)
  sumFi = statTab.returnvaluenumber(statTab.findfield("Sum_Fi"),0)
  if (sumFi = 0)
  then
    meff = 0
    totarea = 0
  else

  if (Fg > 0)
  then meff = sumFi2 / Fg
  else meff = sumFi2 / sumFi
  end
end

```

```

    if (Fg > 0)
        then totarea = Fg
        else totarea = sumFi
    end

    end
    return {meff,areamin,areamax,areaavg,totarea}
else
return {meff,0,0,0,0}
end

```

Subroutine meff_clip (abgeänderte Zeilen des System-Skripts 'GeoProc.Clip.Finish')

```
av.DelayedRun("GeoProc.CloseWait",nil,1)
```

```

theview = av.GetActiveDoc
intheme1 = self.get(0).asstring
intheme2 = self.get(1).asstring
outFName = self.get(2)
select1 = false
select2 = false

```

Subroutine meff_ganz

```

theview = av.GetActiveDoc
themelist = theView.GetThemes
workshp = "$HOME".asfilename.MakeTmp("work", "shp")

```

```

ZgeoTheme = MsgBox.List (themeList, "Select fragmentation polygon layer to be analysed", "Complete fragmentation analysis")
ZgeoTheme.ExportToFTab (workshp)

```

```

Farea = 0
if (MsgBox.YesNo ("Total area equal sum of meshes?", "Complete fragmentation analysis", true) = false) then
    Farea = MsgBox.input("Total area:", "Complete fragmentation analysis", "0").asnumber
end

```

```

meff_exec_return = av.run("meff_exec",{workshp, "", Farea})
meff = meff_exec_return.get(0)
areamin = meff_exec_return.get(1)
areamax = meff_exec_return.get(2)
areaavg = meff_exec_return.get(3)
totarea = meff_exec_return.get(4)

```

```

output = "Effective Mesh Size = "+meff.asstring+NL+NL+
    "Mesh area statistics:"+NL+
    "Minimum = "+areamin.asstring+NL+
    "Maximum = "+areamax.asstring+NL+
    "Mean = "+areaavg.asstring+NL+NL+
    "Total area used: "+totarea.asstring

```

```
msgbox.report( output, "Meff-gesamt (" +ZgeoTheme.asstring+")")
```

```

theproj = av.getproject
theview = av.GetActiveDoc
themelist = theView.GetThemes

```

```

log = av.finddialog("Logtext")
logtext = log.findbyname("logtext")

```

```
ZgeoTheme = MsgBox.List (themeList, "Select a fragmentation polygon layer...", "Analysis of subregions")
```

```

theview.deletetheme(Zgeotheme)
themelist = theView.GetThemes
TgeoTheme = MsgBox.List (themeList, "Analysis for subregions in...", "Analysis of subregions")
theview.addtheme(Zgeotheme)

```

```

ZgeoFtab = ZgeoTheme.getftab
tgeoFtab = tgeotheme.getftab
shpFld = tgeoFtab.findField("Shape")
BT = tgeoFtab.GetSelection
BZ = ZgeoFtab.GetSelection
BTsize = BT.getsize - 1
workshp = "$HOME".asfilename.MakeTmp("work", "shp")

.....

m = { "(1) ...if partially is inside",
      "(2) ...if completely is inside ",
      "(3) ...if center is inside ",
      "(4) ...by cutting-out condition",
      "(5) ...by one-sided relation",
      "(6) ...by two-sided relation"}

m_ausw = MsgBox.choicesasstring( m, "Mesh included in the analysis of a subregion...", "select a
method..." )
Fstat = MsgBox.YesNo ("Calculation of mesh area statistics?", "Analysis of subregions", true)

tgeoFtab.seteditable(true)
mfldnam = "Meff_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
mfld = tgeoFtab.FindField( mfldnam)
if (mfld = nil)
  then mfld = Field.Make(mfldnam, #FIELD_DOUBLE , 16, 8)
  tgeoFtab.addfields({ mfld})
  end

if (fstat = true) then
avefldnam = "MiwFi_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
avefld = tgeoFtab.FindField( avefldnam)
if (avefld = nil) then
  avefld = Field.Make(avefldnam, #FIELD_DOUBLE , 16, 8)
  tgeoFtab.addfields({ avefld})
  end

minfldnam = "MinFi_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
minfld = tgeoFtab.FindField( minfldnam)
if (minfld = nil) then
  minfld = Field.Make(minfldnam, #FIELD_DOUBLE , 16, 8)
  tgeoFtab.addfields({ minfld})
  end

maxfldnam = "MaxFi_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
maxfld = tgeoFtab.FindField( maxfldnam)
if (maxfld = nil) then
  maxfld = Field.Make(maxfldnam, #FIELD_DOUBLE , 16, 8)
  tgeoFtab.addfields({ maxfld})
  end

end

if (fstat = false) then
avefldnam = "MiwFi_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
avefld = tgeoFtab.FindField( avefldnam)
if (avefld <> nil) then
  tgeoFtab.removefields({ avefld})
  end

minfldnam = "MinFi_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
minfld = tgeoFtab.FindField( minfldnam)
if (minfld <> nil) then
  tgeoFtab.removefields({ minfld})
  end

maxfldnam = "MaxFi_"+ZgeoTheme.asstring+"_"+m_ausw.left(3)
maxfld = tgeoFtab.FindField( maxfldnam)

```

```

if (maxfld <> nil) then
  tgeoFtab.removefields({ maxfld})
end

end

.....
*****
if (m_ausw.left(3) = "(1)") then
  log.open
  for each i in 0..Btsize 'rec in tgeoftab

    BT.set(i)
    logtext.setlabel((i+1).asstring)
    tgeoftab.updateselection

    thePoly = tgeoftab.ReturnValue(shpFld,i)
    polyarea = thePoly.returnarea / 1000000

    ZgeoTheme.SelectByTheme (TgeoTheme, #FTAB_RELTYPE_INTERSECTS, 0, #VTAB_SELTYPE_NEW
  )
  ZgeoTheme.ExportToFTab (workshp)

  meff_exec_return = av.run("meff_exec",{workshp, i.asstring, polyarea})
  tgeoFtab.setvalue(mfld, i, meff_exec_return.get(0))
  if (Fstat = true) then
    tgeoFtab.setvalue(minfld, i, meff_exec_return.get(1))
    tgeoFtab.setvalue(maxfld, i, meff_exec_return.get(2))
    tgeoFtab.setvalue(avefld, i, meff_exec_return.get(3))
  end

  BT.clearall
end
log.close
end

*****
if (m_ausw.left(3) = "(2)") then
  log.open
  for each i in 0..Btsize 'rec in tgeoftab

    BT.set(i)
    logtext.setlabel((i+1).asstring)
    tgeoftab.updateselection

    thePoly = tgeoftab.ReturnValue(shpFld,i)
    polyarea = thePoly.returnarea / 1000000

    ZgeoTheme.SelectByTheme (TgeoTheme, #FTAB_RELTYPE_ISCOMPLETELYWITHIN, 0,
#VTAB_SELTYPE_NEW )
    ZgeoTheme.ExportToFTab (workshp)

    meff_exec_return = av.run("meff_exec",{workshp, i.asstring, polyarea})
    tgeoFtab.setvalue(mfld, i, meff_exec_return.get(0))
    if (Fstat = true) then
      tgeoFtab.setvalue(minfld, i, meff_exec_return.get(1))
      tgeoFtab.setvalue(maxfld, i, meff_exec_return.get(2))
      tgeoFtab.setvalue(avefld, i, meff_exec_return.get(3))
    end

    BT.clearall
  end
  log.close
end

*****
if (m_ausw.left(3) = "(3)") then
  log.open
  for each i in 0..Btsize 'rec in tgeoftab

    BT.set(i)

```

```

logtext.setlabel((i+1).asstring)
tgeoftab.updateselection

thePoly = tgeoftab.ReturnValue(shpFld,i)
polyarea = thePoly.returnarea / 1000000

ZgeoTheme.SelectByTheme      (TgeoTheme,      #FTAB_RELTYPE_HASCENTERWITHIN,      0,
#VTAB_SELTYPE_NEW )
ZgeoTheme.ExportToFTab (workshp)
BZ = ZgeoFtab.GetSelection
if (BZ.count > 0)
  then
    meff_exec_return = av.run("meff_exec",{workshp, i.asstring, polyarea})
    tgeoFtab.setvalue(mfld, i, meff_exec_return.get(0))
    if (Fstat = true) then
      tgeoFtab.setvalue(minfld, i, meff_exec_return.get(1))
      tgeoFtab.setvalue(maxfld, i, meff_exec_return.get(2))
      tgeoFtab.setvalue(avefld, i, meff_exec_return.get(3))
    end
  else
    tgeoFtab.setvalue(mfld, i, 0)
    if (Fstat = true) then
      tgeoFtab.setvalue(minfld, i, 0)
      tgeoFtab.setvalue(maxfld, i, 0)
      tgeoFtab.setvalue(avefld, i, 0)
    end
  end
  BT.clearall
end
log.close
end

*****
if (m_ausw.left(3) = "(4)") then

  log.open
  for each i in 0..BTsize

    BT.set(i)
    logtext.setlabel((i+1).asstring)

    tgeoftab.updateselection

    thePoly = tgeoftab.ReturnValue(shpFld,i)
    polyarea = thePoly.returnarea / 1000000

    clipshp = "$HOME".asfilename.MakeTmp("clip", "shp")
    av.run("meff_clip",{ZgeoTheme, tgeotheme, clipshp})
    clipgeotheme = theme.make(srcname.make(clipshp.asstring))
    clipgeoTheme.ExportToFTab(workshp)

    if (file.canDelete(clipshp)) then
      file.delete(clipshp)
    end

    meff_exec_return = av.run("meff_exec",{workshp, i.asstring, polyarea})
    tgeoFtab.setvalue(mfld, i, meff_exec_return.get(0))
    if (Fstat = true) then
      tgeoFtab.setvalue(minfld, i, meff_exec_return.get(1))
      tgeoFtab.setvalue(maxfld, i, meff_exec_return.get(2))
      tgeoFtab.setvalue(avefld, i, meff_exec_return.get(3))
    end

    BT.clearall
  end
log.close

```

```

end

*****
if (m_ausw.left(3) = "(5)") then

    z_workshp = "$HOME".asfilename.MakeTmp("z_work", "shp")
    ZgeoTheme.ExportToFTab(z_workshp)
    vt = ftab.make(srcname.make(z_workshp.asstring))
    vt.seteditable(true)
    ferg = vt.FindField("Ferg")           'Fläche vor clip
    if (ferg = nil)
        then ferg = Field.Make("Ferg", #FIELD_DOUBLE, 16, 2)
        vt.addfields({ ferg})
    end
    calcstr = "[Shape].ReturnArea / 1000000"
    vt.Calculate( calcstr, ferg)
    vt.seteditable(false)

    z_workshptheme = theme.make(srcname.make(z_workshp.asstring))
    av.GetActiveDoc.addtheme(z_workshptheme)

    log.open
    for each i in 0..BTsize

        BT.set(i)
        logtext.setlabel((i+1).asstring)

        tgeoftab.updateselection

        thePoly = tgeoftab.ReturnValue(shpFld,i)
        polyarea = thePoly.returnarea / 1000000

        clipshp = "$HOME".asfilename.MakeTmp("clip", "shp")
        av.run("meff_clip",{z_workshptheme, tgeotheme, clipshp})
        clipgeotheme = theme.make(srcname.make(clipshp.asstring))
        clipgeoTheme.ExportToFTab(workshp)

        if (file.canDelete(clipshp)) then
            file.delete(clipshp)
        end

        meff_exec_return = av.run("meff_exec_2",{workshp, i.asstring, polyarea, "(5)"}

        tgeoftab.setvalue(mfld, i, meff_exec_return.get(0))
        if (Fstat = true) then
            tgeoftab.setvalue(minfld, i, meff_exec_return.get(1))
            tgeoftab.setvalue(maxfld, i, meff_exec_return.get(2))
            tgeoftab.setvalue(avefld, i, meff_exec_return.get(3))
        end

        BT.clearall
    end
    log.close

    if (file.canDelete(clipshp)) then
        file.delete(clipshp)
    end

    z_workshptheme.getftab.deactivate
    av.GetActiveDoc.deletetheme(z_workshptheme)
    if (file.canDelete(z_workshp)) then
        file.delete(z_workshp)
    end
end

*****
if (m_ausw.left(3) = "(6)") then

```

```

z_workshp = "$HOME".asfilename.MakeTmp("z_work", "shp")
ZgeoTheme.ExportToFTab(z_workshp)
vt = ftab.make(srcname.make(z_workshp.asstring))
vT.seteditable(true)
ferg = vT.FindField( "Ferg")           Fläche vor clip
if (ferg = nil)
  then ferg = Field.Make("Ferg", #FIELD_DOUBLE , 16, 2)
  vT.addfields({ ferg})
end
calcstr = "[Shape].ReturnArea / 1000000"
vT.Calculate( calcstr, ferg)
vt.seteditable(false)

z_workshptheme = theme.make(srcname.make(z_workshp.asstring))
av.GetActiveDoc.addtheme(z_workshptheme)

log.open
for each i in 0..BTsize

  BT.set(i)
  logtext.setlabel((i+1).asstring)

  tgeoftab.updateselection

  thePoly = tgeoftab.ReturnValue(shpFld,i)
  polyarea = thePoly.returnarea / 1000000

  clipshp = "$HOME".asfilename.MakeTmp("clip", "shp")
  av.run("meff_clip",{z_workshptheme, tgeotheme, clipshp})
  clipgeotheme = theme.make(srcname.make(clipshp.asstring))
  clipgeotheme.ExportToFTab(workshp)

  meff_exec_return = av.run("meff_exec_2",{workshp, i.asstring, polyarea, "(6)")

  tgeoftab.setvalue(mfld, i, meff_exec_return.get(0))
  if (Fstat = true) then
    tgeoftab.setvalue(minfld, i, meff_exec_return.get(1))
    tgeoftab.setvalue(maxfld, i, meff_exec_return.get(2))
    tgeoftab.setvalue(avefld, i, meff_exec_return.get(3))
  end

  BT.clearall
end
log.close

if (file.canDelete(clipshp)) then
  file.delete(clipshp)
end

z_workshptheme.getftab.deactivate
av.GetActiveDoc.deletetheme(z_workshptheme)
if (file.canDelete(z_workshp)) then
  file.delete(z_workshp)
end

end

.....
tgeoftab.seteditable(false)

if (file.canDelete(workshp)) then
  file.delete(workshp)
end

msgbox.info("Finished!", "meff_tool")

```