



WUFI®

Multicore und Batch-Berechnung mit WUFI® 2D

Stand: August 2026

Problematik.....	<u>Folie 3</u>
Lösungsansatz.....	<u>Folie 6</u>
Verwendung der Skripte.....	<u>Folie 7</u>
WUFI2D_BatchCalc	<u>Folie 10</u>
WUFI2D_CoreTest	<u>Folie 24</u>
WUFI2D_StartGUI	<u>Folie 35</u>

Kernproblem:

Auf neueren CPUs mit vielen Kernen ist WUFI 2D teilweise langsamer als auf älteren CPUs mit bis zu vier Kernen.

Ursache:

- Um mehrere Kerne zu nutzen, muss das zu lösende Differentialgleichungssystem aufgeteilt werden.
- Zwischen den genutzten Kernen müssen dabei Informationen ausgetauscht werden, was Zeit kostet (sogenannter Synchronisations-Overhead).
- Je mehr Kerne genutzt werden, desto größer wird dieser Overhead und desto länger dauert die Berechnung.

Verhalten von WUFI 2D:

- Standardmäßig nutzt WUFI 2D für die Berechnung alle vorhandenen CPU-Kerne (inklusive Hyperthreading) bis auf einen; der verbleibende Kern wird für die Benutzeroberfläche verwendet.
- Abhängig von der Konstruktion ist meist eine Anzahl von etwa 4 bis 10 Kernen optimal. Werden mehr Kerne verwendet, nimmt die Berechnungsgeschwindigkeit wieder ab.
- Die Anzahl der genutzten Kerne kann derzeit nicht direkt in WUFI 2D eingestellt werden.

Vergleiche mit unterschiedlicher Anzahl von Kernen:

„Bed-Plate“ Benchmark - Berechnung über 3 Jahre (26280 h)

CPU	System	Physische Kerne	Logische Kerne	verwendet	Berechnungszeit
Intel i9-11900K	Win 10	16	32	15	9,5 h
Intel i9-11900K	Win 10	16	32	4	1,6 h
Intel i9-14900	Win 10	24	32	31	6,3 h
Intel i9-14900	Win 10	24	32	8	4,8 h

Da WUFI 2D derzeit weder die direkte Vorgabe der Kernanzahl noch die automatisierte parallele Berechnung mehrerer Simulationen unterstützt, wurden hierfür PowerShell-basierte Skripte entwickelt:

- **WUFI2D_BatchCalc:**
Ermöglicht die Berechnung ein oder mehrerer vorbereiteter WUFI 2D-Projektdateien im Batch-Modus mit der optimalen Anzahl von Kernen über eine Berechnungsliste.
- **WUFI2D_CoreTest:**
Ermittelt die optimale Anzahl an Kernen (Threads) für System und Projekt.
- **WUFI2D_StartGUI:**
Startet die WUFI 2D-Oberfläche mit einer benutzerdefinierten Anzahl an Kernen.
- **WUFI2D_StartGUI_fastlane:**
Startet die WUFI 2D-Oberfläche direkt mit der zuletzt benutzten Anzahl an Kernen.
Kann zum regulären Start des Programms verwendet werden.

Die aktuellen Versionen der Skripte sowie weitere Informationen zu den Skripten befinden sich im WUFI-Forum: <https://www.wufi-forum.com/viewtopic.php?t=2357>

Wichtige Dateien im WUFI-Projektverzeichnis:

- *wufi2d.xml* – Projektdefinition(*)
- *input.dat* – Steuerdatei Rechenkern(*)
- *<Projektname>.wfd* – Ergebnisdatei

(*): muss mindestens enthalten sein

Weitere Dateien - zur Info:

- *iter.txt* – Rechnungsablauf
- *input.bak* – Sicherung der *input.dat* bei Änderung durch BatchCalc
- **.bcli* – Klimarandbedingungen
- *<material>.xml* – Materialdaten

Vorbereitung der WUFI 2D-Projekte:

- Einrichten eines jeden Projekts in der grafischen Benutzeroberfläche
- Wechseln zum Reiter „Berechnung“ – dadurch wird die Eingabedatei (*input.dat*) erzeugt.
- Starten der Berechnung („Anrechnen“)
- Abbrechen der Berechnung nach ein paar Zeitschritten und Speichern des Projekts.
→ Alle erforderlichen Dateien werden im Projektordner abgelegt.

Voraussetzungen:

- Betriebssystem: Windows 11, PowerShell 5.1 oder höher
- Software: Installation von WUFI 2D 4.x
- Kernel-Pfad:
C:\Program Files (x86)\WUFI\WUFI2D-4\kernel\x64\w2dcore-x64.exe

Verwendung der Skripte

Starten der Skripte:

1. Alle Dateien/ Ordner aus dem ZIP-Archiv in einen Ordner entpacken.
2. Die Verknüpfungen können per Doppelklick gestartet werden:
 - *WUFI2D_BatchCalc.Ink*
 - *WUFI2D_CoreTest.Ink*
 - *WUFI2D_StartGUI.Ink*
 - *WUFI2D_StartGUI_fastlane.Ink*

Hinweise:

- Die Anzahl der Kerne wird ausgelesen und angezeigt.
- Es kann jeweils nur eine Instanz der Skripte geöffnet werden.
- Die Skripte können mit <Strg + C> abgebrochen werden

Konvention:

Calc-Kerne = *OMP-Kerne minus 1* (OMP = Windows Systemvariable: Open Multi-Processing)
Bei Abfragen in den Skript-Oberflächen sind immer die Kerne für die Berechnung (*Calc-Kerne*) anzugeben.

WUFI2D_BatchCalc

Mit diesem Skript können ein oder mehrere WUFI 2D-Projekte automatisch im Hintergrund (batch-Betrieb) nacheinander berechnet werden.

Kernfunktionen:

- Eingabe der für die Berechnung verwendete Kernanzahl.
- Es können mehrere Projekte vorbereitet und nacheinander berechnet werden.
- Die Einstellungen werden für jedes Projekt gespeichert (*batchcalc.xml*), damit diese auch bei späteren Berechnungen wieder zur Verfügung stehen.
- Die vorbereitete Liste mit Projekten wird ebenfalls gespeichert (*batch-prepare.xml*) und kann später wieder geladen / bearbeitet werden.
- Der Berechnungszeitraum eines bereits berechneten Projekts kann verlängert werden („Weiterrechnen“).

Hinweis: Der Batch-Modus verwendet direkt das Projektverzeichnis. Daher hat auch die Schreibgeschwindigkeit einen gewissen Einfluss. Um optimale Geschwindigkeiten zu erreichen, sollten die Projekte für die Verwendung mit BatchCalc auf einer lokalen Festplatte liegen.

Neu beginnen oder Projektliste verwenden

Existiert eine gespeicherte Liste (*batch-prepare.xml*), zeigt das Skript diese an und fragt:

```
[1] Use saved list  
[2] Start fresh  
Select (1/2) [1]:
```

Eingabe	Bedeutung
1 oder Enter	Gespeicherte Liste verwenden
2	Neu beginnen

Projekte auswählen

Please select a WUFI 2D project directory

→ Auswahl des Projektordners über Windows-Dialog.

Ist das Projekt bereits in der Liste:

Already in queue.

[1] Reconfigure

[2] Keep

(1/2) [2]:

Eingabe	Bedeutung
1	Neu konfigurieren
2 oder Enter	Bestehende Einstellung behalten

Berechnungsstunden festlegen

Hier kann der Berechnungszeitraum nochmal geändert werden, wenn gewünscht:

Calculation hours (Enter = keep <Wert> h):

Eingabe	Bedeutung
Enter	Aktuelle Stundenzahl beibehalten
Zahl	Neue Stundenzahl setzen

Anzahl Kerne festlegen

Fall A – gespeicherte Informationen vorhanden
(*core-test.xml* oder *batchcalc.xml*)

```
core-test.xml: Calc cores: X (OMP=X+1)  
Use these calc cores? (Enter = X, or enter value, max <max>):
```

Eingabe	Bedeutung
Enter	Gespeicherte Rechenkerne übernehmen
Zahl	Eigene Rechenkernzahl setzen

Anzahl Kerne festlegen

Fall B – keine gespeicherten Informationen vorhanden oder auf einem anderen Computer erzeugt.

Zuerst fragt das Skript, ob ein Core-Test ausgeführt werden soll:

```
No core-test.xml found.
```

```
(bzw.: core-test.xml from different CPU!)
```

```
Run CoreTest now? (Y/N) [Y]:
```

Eingabe	Bedeutung
Enter oder Y	Core-Test wird jetzt ausgeführt
N	Kein Core-Test → weiter zur manuellen Eingabe (siehe nächste Folie)

Anzahl Kerne festlegen

Wird kein Core-Test ausgeführt – manuelle Eingabe:

```
CPU: <Name> (max <maxCalc> calc cores)
```

```
Enter calc cores (1-<maxCalc>):
```

Eingabe	Bedeutung
Zahl 1 ... maxCalc	Rechenkerne festlegen
Zahl > maxCalc	Kernanzahl wird auf Maximum begrenzt

Berechnungsmodus (Restart)

Erscheint nur, wenn folgende Bedingungen erfüllt sind:

- eine Ergebnisdatei (*.wfd*) existiert bereits und
- die neu gewählte Stundenzahl ist größer als die im Projekt gespeicherte

```
Result file found: <Datei>
```

```
[1] Run full calculation (<neu> hours)
```

```
[2] Resume calculation (from hour <alt> to hour <neu>)
```

```
Mode (1/2) [1]:
```

Eingabe	Bedeutung
1 (oder Enter)	Komplette Neuberechnung über die volle Stundenzahl
2	Fortsetzen ab der letzten Stunde (from hour <alt> to hour <neu>)

Ergebnisdatei umbenennen

Bei der Batch-Berechnung wird die Ergebnisdatei nicht automatisch wie das Projekt benannt.

```
The result filename will be 'wufi2dresult.wfd' (WUFI default).  
Rename result file to '<Projektname>.wfd' after calculation?  
Rename? (Y/N) [Y]:
```

Eingabe	Bedeutung
Enter oder Y	Ergebnisdatei wird nach der Berechnung in <i><Projektname>.wfd</i> umbenannt
N	Standardname <i>wufi2dresult.wfd</i> bleibt erhalten

Aktionsmenü

BATCH PREPARATION STATUS

[1] Start calculation (all prepared projects)

[2] Add another project

[3] Edit project settings -> reconfiguration

[4] Delete a project from list

[5] Delete project list

[6] Exit without calculating

Select (1-6) [1]:

Eingabe	Bedeutung
1 oder Enter	Berechnung aller Projekte starten
2	Weiteres Projekt hinzufügen
3	Projekteinstellungen bearbeiten
4	Projekt aus Liste löschen
5	Gesamte Projektliste löschen
6	Beenden ohne Berechnung

Berechnung läuft

- Projekte werden nacheinander berechnet
- Fortschrittsanzeige: Time step X / Y (Z%) | Elapsed: hh:mm:ss

```
Batch calculation (2 project(s))
Project 1 of 2 : Variante7_Start0kt
[
Calculating: Variante7_Start0kt
Time step 6705 / 43800 (15%) | Elapsed: 00:14:09
[ooooooooooooooooooooo
```

Zusammenfassung

Nach Abschluss aller Berechnungen wird eine Zusammenfassung angezeigt:

```
=====
ALL CALCULATIONS COMPLETED
Total: hh:mm:ss
=====
[OK]    <Projekt>
        Wallclock: hh:mm:ss | CPU: <s>s | Calc cores: X (OMP=Y)
[??]    <Projekt>
        Wallclock: ... | CPU: ... | Calc cores: X (OMP=Y)
[FAIL]  <Projekt>: <Fehlermeldung>
Summary: <n> OK, <m> failed
```

Symbol	Bedeutung
[OK]	Berechnung erfolgreich beendet
[??]	Berechnung durchgelaufen, keine Rückmeldung vom Rechenkern
[FAIL]	Berechnung fehlgeschlagen (Fehlermeldung wird angezeigt)

Zusammenfassung

Folgende Abfrage erscheint nur, wenn kein Projekt fehlgeschlagen ist:

```
Delete batch-prepare.xml? (Y/N) [Y]:
```

Eingabe	Bedeutung
Enter oder Y	<i>batch-prepare.xml</i> wird gelöscht
N	Liste bleibt erhalten und kann später erneut geladen werden

WUFI2D_CoreTest

Das Skript ermittelt automatisch die optimale Anzahl an Kernen für eine WUFI 2D-Berechnung. Dazu wird derselbe Rechenfall mit unterschiedlicher Kernanzahl wiederholt, bis das Optimum ermittelt ist.

Hinweise:

- Der Berechnungszeitraum für den Test kann angegeben werden (empfohlen: 730 h).
- Der Startmonat kann gesetzt werden (empfohlen: kältester Monat).
- Ergebnis wird in das Projektverzeichnis gespeichert (*core-test.xml*).

Hinweis:

Im Batch-Modus wird direkt das Projektverzeichnis zur Berechnung benutzt. Daher hat auch die Schreibgeschwindigkeit einen gewissen Einfluss. Um optimale Geschwindigkeiten zu erreichen, sollten die Projekte für den Core-Test auf einer lokalen Festplatte liegen.

Schritt 1: Projektverzeichnis auswählen

Ist der aktuelle Ordner kein gültiges WUFI-Projekt, öffnet sich ein Auswahldialog:

```
Current directory is NOT a WUFI 2D project directory.  
Please select the WUFI 2D project directory...
```

Bei ungültiger Auswahl:

```
Selected directory is NOT valid!  
Try again? [Y/N] (Default: Y):
```

Eingabe	Bedeutung
Enter oder Y	Erneut ein Verzeichnis auswählen
N	Abbruch des Skripts

Schritt 2: Vorhandenes Ergebnis prüfen

Sind gespeicherte Informationen vorhanden (*core-test.xml*), zeigt das Skript deren Inhalt an und fragt:

Fall A – Core-Test wurde auf derselben CPU durchgeführt:

```
Run a new test? [Y] Yes / [N] No, keep existing result  
(Default: N):
```

Eingabe	Bedeutung
Enter oder N	Vorhandenes Ergebnis behalten, Skript endet
Y	Neuen Test starten

Schritt 2: Vorhandenes Ergebnis prüfen

Sind gespeicherte Informationen vorhanden (*core-test.xml*), zeigt das Skript deren Inhalt an und fragt:

Fall B – Core-Test wurde auf einer anderen CPU durchgeführt:

```
The existing result was calculated on a different CPU.  
Run a new test? [Y] Yes (recommended) / [N] No, keep existing  
result (Default: Y):
```

Eingabe	Bedeutung
Enter oder Y	Neuen Test starten (empfohlen)
N	Vorhandenes Ergebnis behalten

Schritt 3: Physische oder logische Kerne testen

CPU cores detected:

[1] Physical cores: <n>

[2] Logical cores: <m> (incl. Hyper-Threading)

Which cores should be tested? [1] Physical / [2] Logical
(Default: 2):

Eingabe	Bedeutung
Enter oder 2	Logische Kerne testen (Empfehlung)
1	Physische Kerne testen

Schritt 4: Startmonat festlegen

Current start date in file: 1.10.1991_0.00

Recommendation: choose the coldest month of the year.
Enter 'K' to keep the start date from input.dat.

Which start month? (1-12, [K] keep from input.dat, Default: 1 = January):

Eingabe	Bedeutung
Enter	Standardwert 1 (= Januar) verwenden
Zahl 1 bis 12	Entspricht dem Monat, auf den das Startdatum gesetzt werden soll. Der Starttag wird immer auf Tag 1 festgelegt.
K	Startdatum aus <i>input.dat</i> verwenden

Schritt 5: Test-Stundenzahl festlegen

Recommended: 730 hours for test.

How many hours for the test? (Default: 730):

Eingabe	Bedeutung
Enter	Standardwert 730 h verwenden
Zahl > 0	Eigene Test-Stundenzahl setzen

Schritt 6: Test starten

Press Enter to start the core test

Eingabe	Bedeutung
Enter	Testläufe starten

Anschließend läuft der Test automatisch:

- Fortschrittsbalken pro Zeitschritt:

```
WUFI 2D Core Test
OMP=8 (calc cores: 7)...
[ooooooooooooooooooooo]
Calculating: OMP=8 (calc cores: 7)
Time step 360 / 730 (49%) | Elapsed: 00:00:57
[ooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo]
```

- Automatischer Abbruch nach zwei aufeinanderfolgenden Berechnungsläufen ohne Verbesserung

Schritt 7: Ergebnis und Speichern

Nach Abschluss zeigt das Skript eine Ergebnistabelle:

CalcCores	OMP	CPU Time (s)	Wallclock (s)	Note
-----	---	-----	-----	----
...				
★ Optimal: Calc cores: X (OMP_NUM_THREADS=X+1)				

- Der optimale Wert wird mit << optimal markiert.

```
Results:
=====
Project: C:\DATEN\WUFI2D\BedPlate_Benchmark
Grid: 152 * 202 | Test: 730 h | Original: 26280 h

CalcCores  OMP  CPU Time (s)  Wallclock (s)  Note
-----
7          8    203.4        217.1
8          9    151.0        166.0
9         10    144.1        158.2
10        11    141.9        156.1      << optimal
11        12    145.9        160.3
12        13    145.7        160.4

★ Optimal: Calc cores: 10 (OMP_NUM_THREADS=11)

Total duration: 00:17:00
Results saved:
C:\DATEN\WUFI2D\BedPlate_Benchmark\core-test.xml
```

- Ergebnis wird automatisch in *core-test.xml* gespeichert.

Schritt 8: Weiteres Projekt testen

Do you want to test another project? [Y] Yes / [N] No
(Default: N):

Eingabe	Bedeutung
Enter oder N	Skript beenden
Y	Weiteres Projekt testen (zurück zu Schritt 1)

WUFI2D_StartGUI

Das Skript startet die WUFI 2D-Benutzeroberfläche mit einer vorgegebenen Anzahl an Kernen.

Funktionen:

- Die Anzahl der Kerne kann angegeben werden.
- Ergebnisse eines Core-Tests können aus einem Projektverzeichnis geladen werden.
- Die angegebene Kernanzahl wird gespeichert und bei erneutem Öffnen vorgeschlagen.
- Direktes Öffnen von WUFI 2D unter Verwendung der gespeicherten Kerne ohne weitere Rückfragen (*WUFI2D_StartGUI_fastlane*).

Möglichkeiten zum Angeben / Laden der Kernanzahl

Das Skript bietet verschiedene Möglichkeiten, die Kernanzahl zu setzen:

Situation	Verhalten
Manuelle Eingabe	Direkte Eingabe der Kernanzahl
Gespeicherter Wert	Gespeicherter Wert des letzten Aufrufs wird angeboten
Laden aus einem Projekt	Auswahl eines Projektverzeichnis

Manuelle Eingabe / Gespeicherten Wert verwenden

Liegt eine gespeicherte Konfiguration vor, wird diese angezeigt. Andernfalls ist nur die manuelle Eingabe oder das Laden aus einem Projektverzeichnis verfügbar.

```
Use X calc cores? [Enter]
Enter new number of calc cores (2-<max>)
load from a WUFI 2D project [P]
[Enter] / (2-<max>) / [P]
```

Eingabe	Bedeutung
Enter	Übernimmt den gespeicherten Wert
Zahl 4 - <max>	Neue Anzahl an Kernen manuell festlegen
P	Kernanzahl aus einem WUFI 2D-Projekt laden

Auswahl Projektverzeichnis (bei Option [P])

Ermöglicht das Laden eines Core-Tests aus einem WUFI 2D-Projektverzeichnis.

Please select a project directory containing a previous
core-test.xml...

Eingabe	Bedeutung
Ordner wählen	Liest die <i>CoreTest-Daten</i> aus dem gewählten Verzeichnis
Abbrechen	Bricht die Auswahl ab

Nach dem Laden wird die Kernanzahl angezeigt:

Use X calc cores? [Enter] = Yes / or enter number of calc cores
(4-<max>)

Eingabe	Bedeutung
Enter	Übernimmt den geladenen Wert aus dem Projekt
Zahl 4 - <max>	Eigenen Wert festlegen

WUFI 2D starten

Das Skript prüft die gewählte Anzahl an Kernen und setzt diese für den Start von WUFI 2D:

```
=====
Starting WUFI 2D with <X> calculation cores...
(OMP_NUM_THREADS=<X+1>)
=====
```

WUFI 2D wird gestartet und das Skript beendet.