



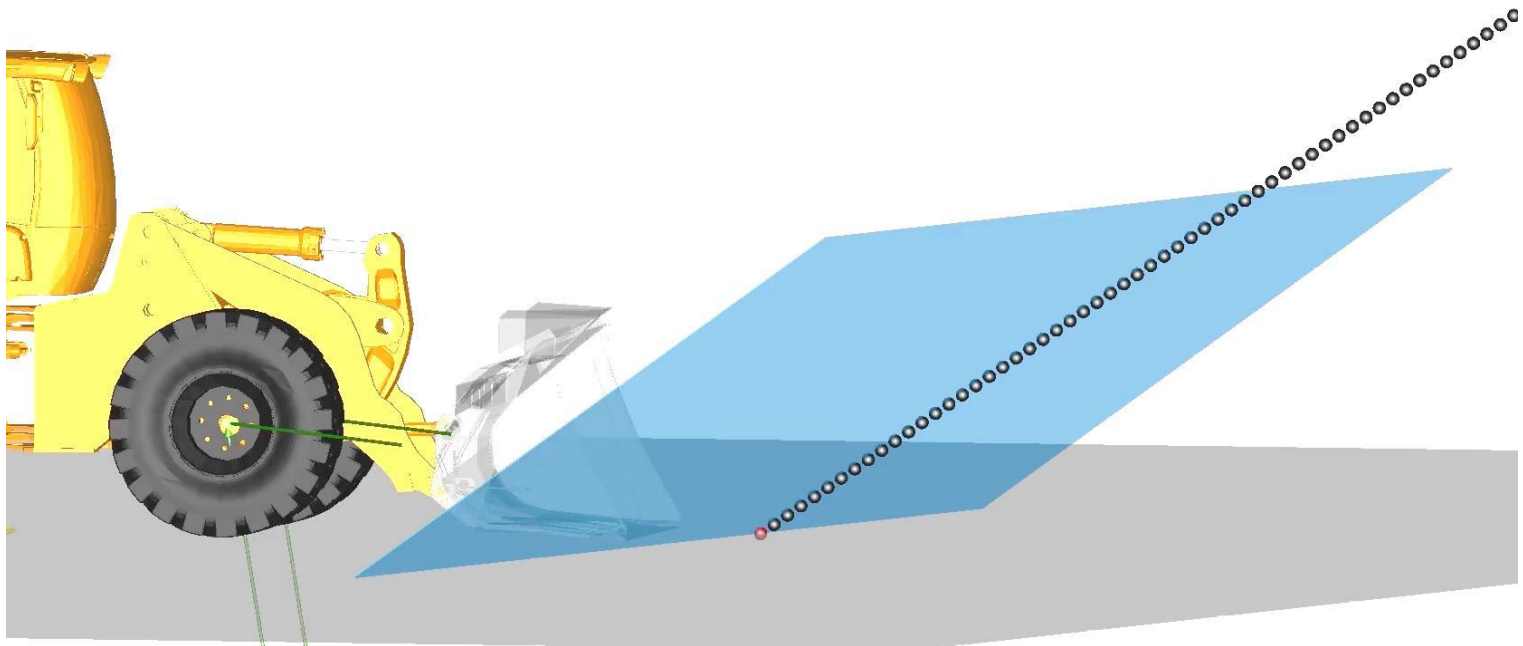
# Minimalwertbildung mit Modelica – Externe Speichererweiterung

**„Nature doesn't know or care about  
time steps.. ..  
but simulation does.“**

Dipl.-Ing. Georg Heß  
Modelisax-Treffen



Ziel: Widerstandmodell in Form eines Haufwerks

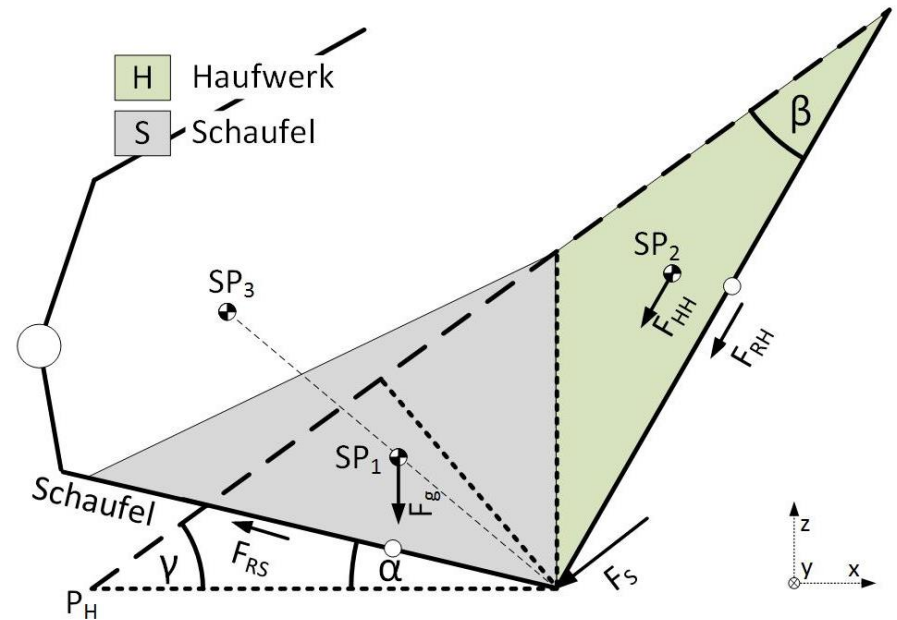
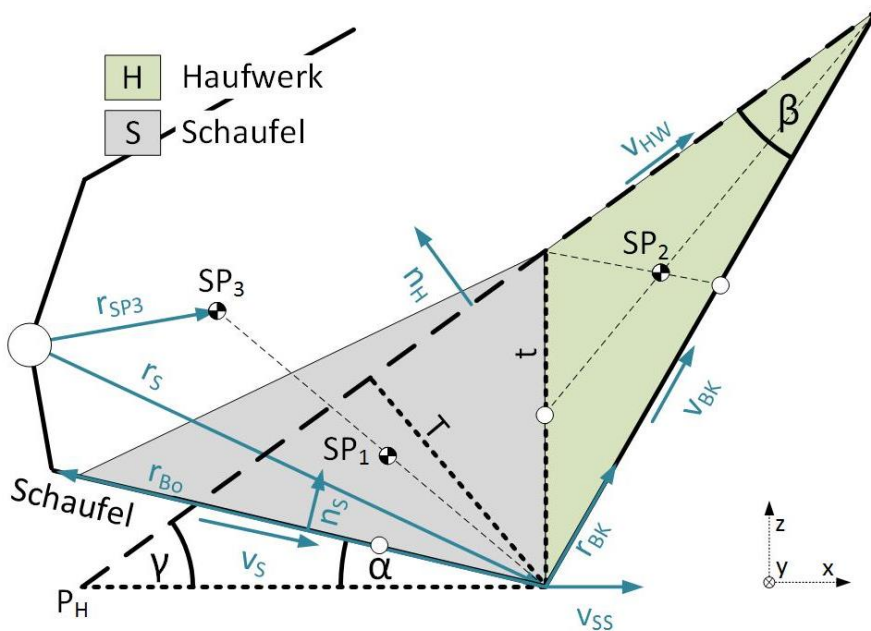


Ziel: Widerstandmodell in Form eines Haufwerks

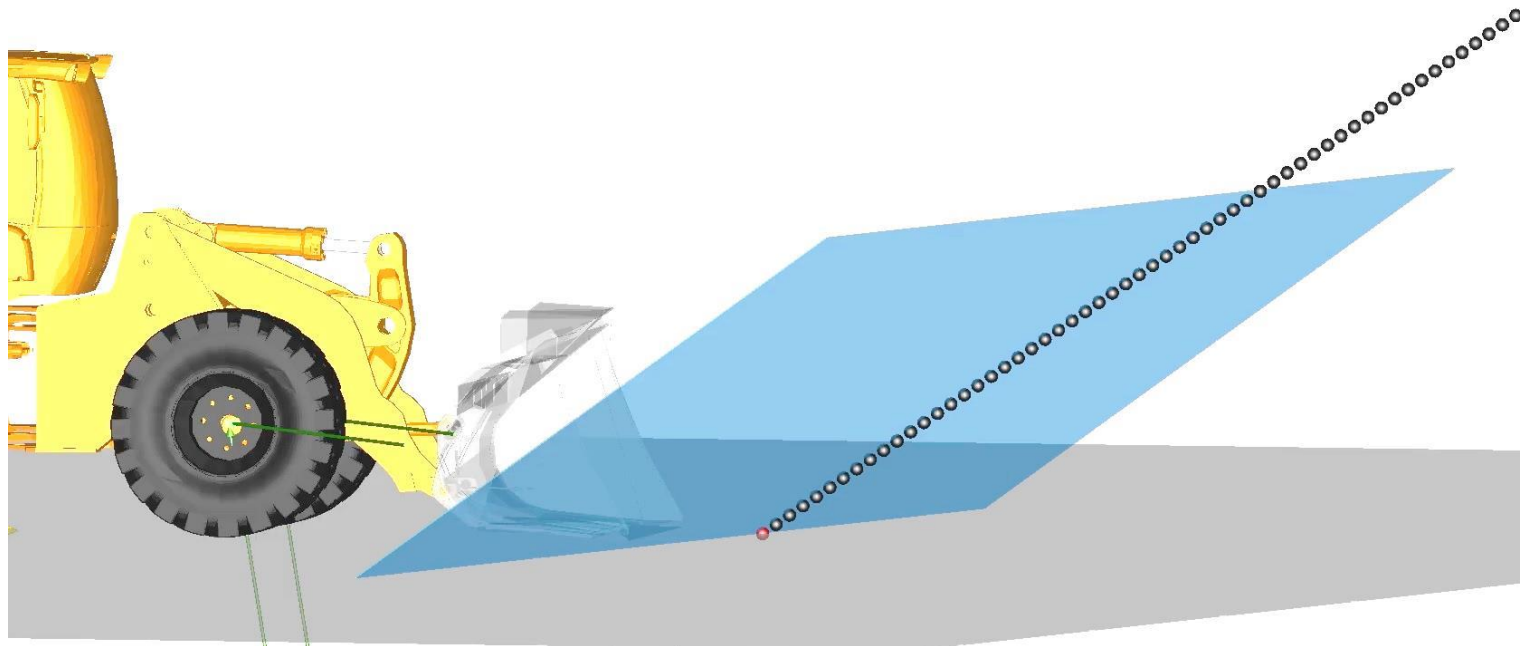
$$A_H = \frac{t^2 (\cos(\gamma + \beta))}{2 \sin(\beta)} \cos(\gamma)$$

$$F_S = aT^b$$

$$A_S = \int tv_{SS} d\tau$$



Ziel: Widerstandmodell in Form eines Haufwerks

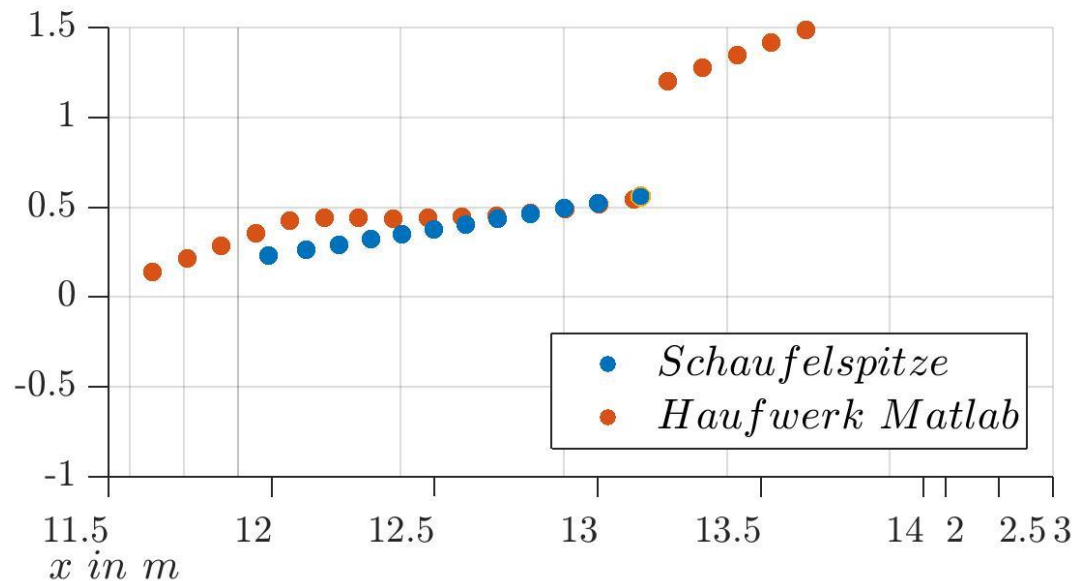


Ziel: Widerstandmodell in Form eines Haufwerks

Bodenkraft auf Schaufelboden

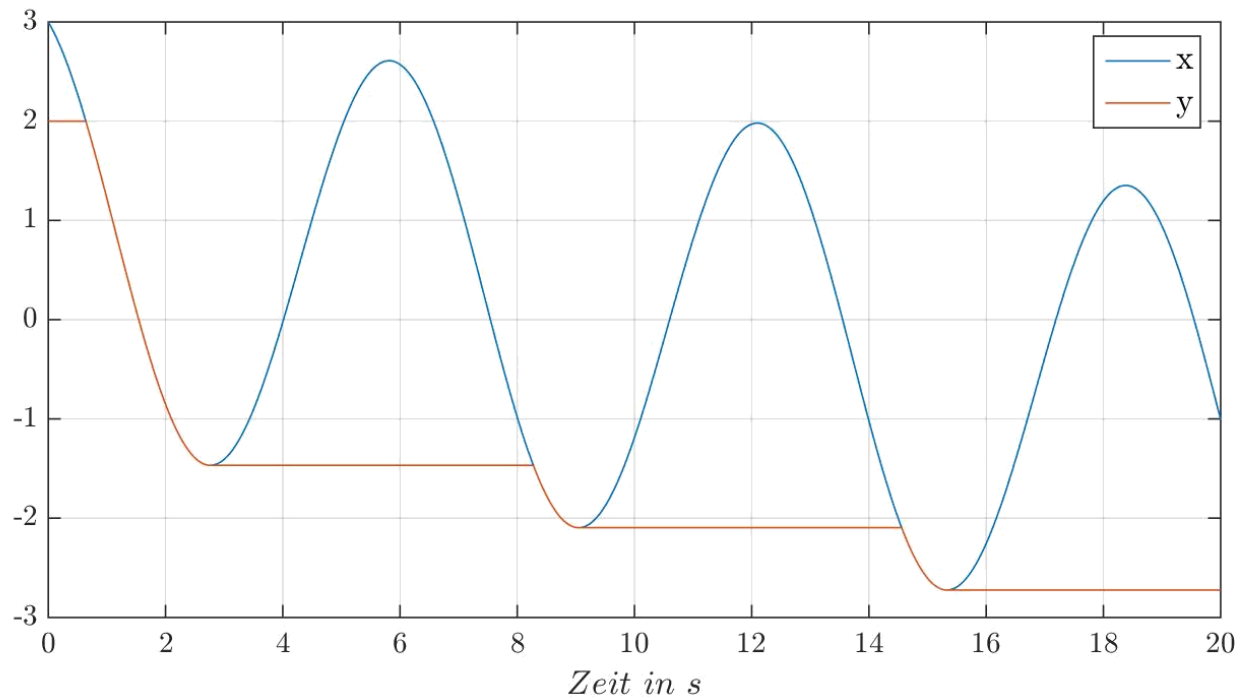
Lsg: Schnittkontur erzeugen

→ Oberfläche höher als Schaufelspitze → Schnitt



## Minimalwertbildung siehe 5. Modelisax-Treffen (Dr. Clauß)

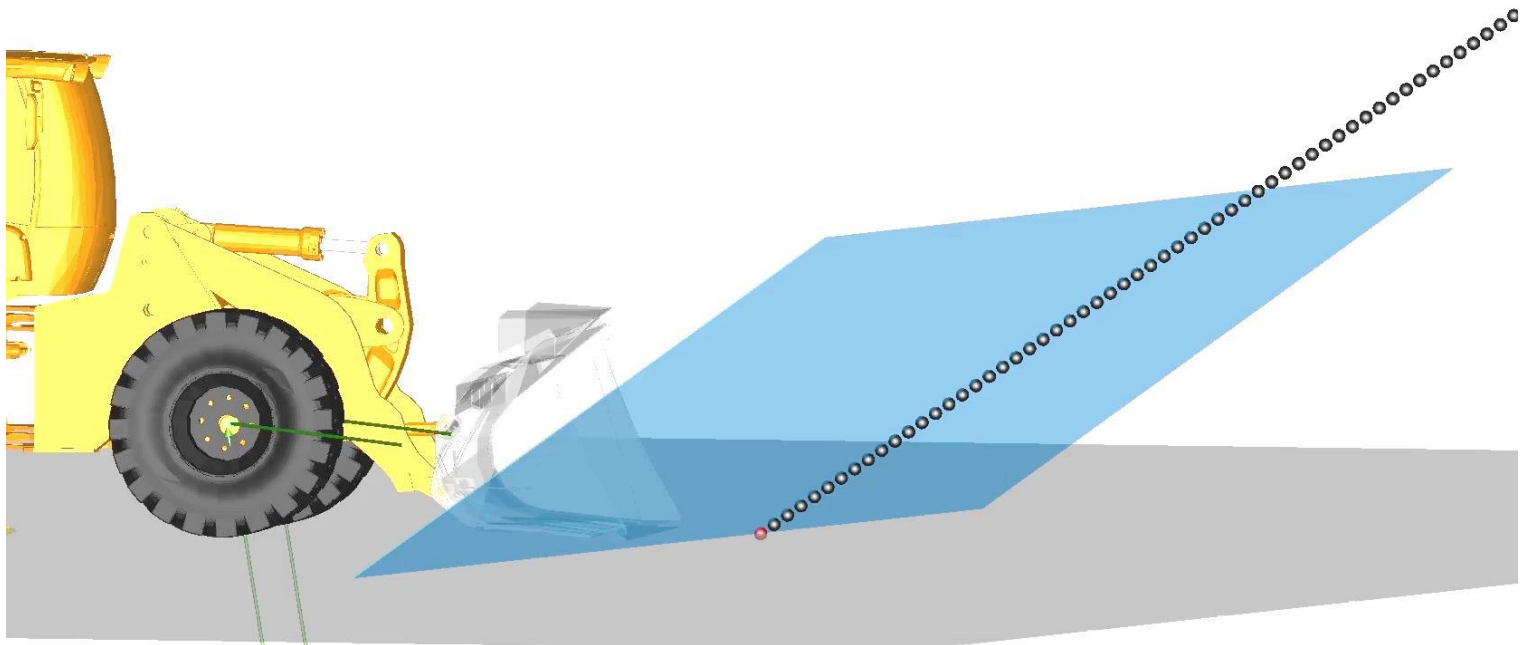
- Geg:  $x(t)$
- Ges:  $y(t) = \min(x(t))$
- Lsg:



**→In Modelica Problembehaftet**

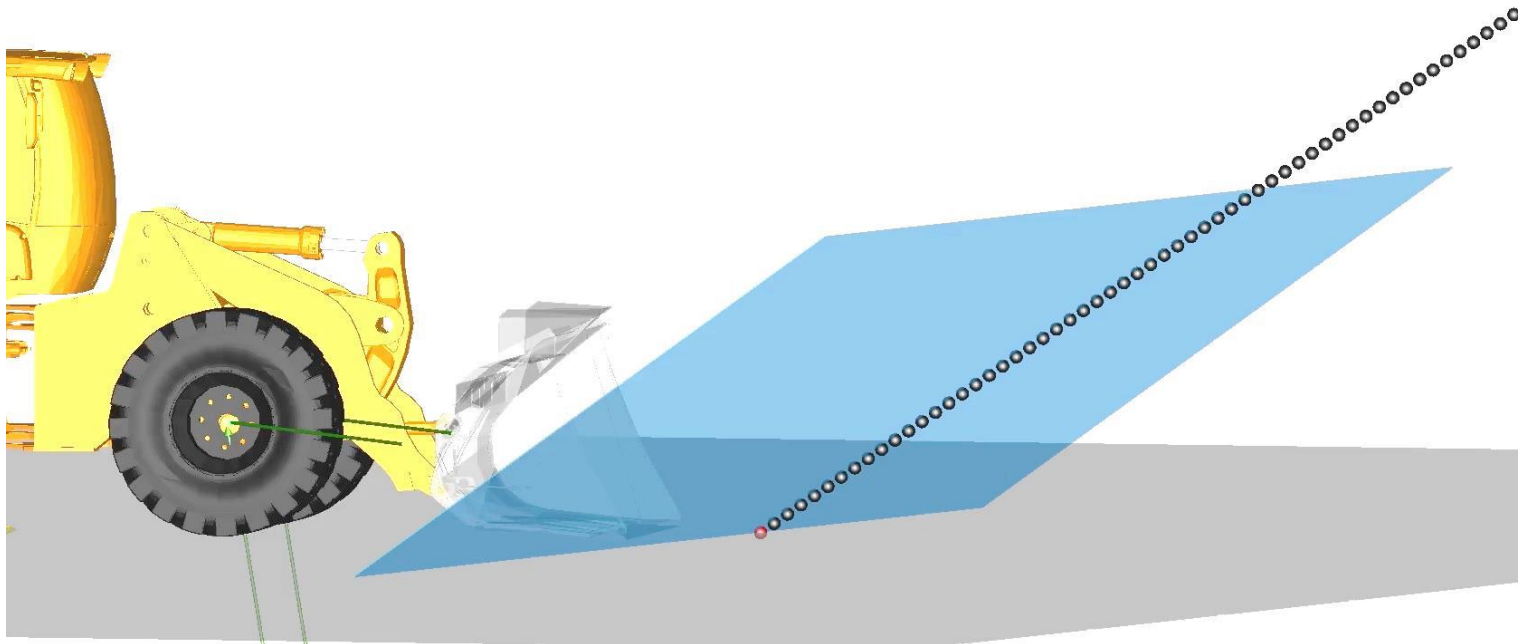
## Berechne die Verschiebung der Punkte mit einem Algorithmus

- $(dz, \dots) = \text{Bodendurchdringung}(dz, \dots)$
- Haufwerksberechnung **ohne** Berechnung der Bodenkraft
  - Nichtlineares Gleichungssystem (Größe = Anzahl der Haufwerkpunkte)



## Berechne die Verschiebung der Punkte mit einem Algorithmus

- $(dz, \dots) = \text{Bodendurchdringung}(dz, \dots)$
- Haufwerksberechnung **mit** Berechnung der Bodenkraft + Kopplung
  - Nichtlineares Gleichungssystem (Größe = Anzahl der Haufwerkpunkte)

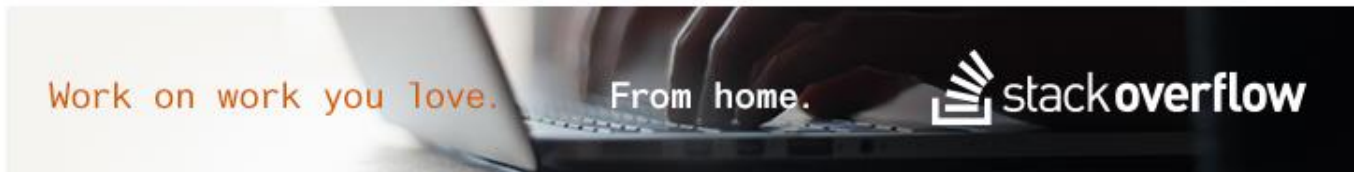




- Direkte Zuweisung  $f(t) = f(t - 1)$  nicht möglich!
- Sample() und pre() Operator in when Klausel möglich
  - Kein nichtlineares GS mehr
  - Events

... Erkenntnisse von Dr. Clauß

- „Es müsste in Modelica was geben, was die Werte aus vorangegangenen Zeitschritten bereitstellt!“



- ▲ Is it possible to use the previous value of the time varying variable
- ▲ The timestep does not enter into it at all. A model that uses information about timestep is just wrong. Nature doesn't know or care about integration time steps, the model should reflect that.

**„Geschehenes läßt sich nicht ungeschehen machen.“**  
— **Marcus Tullius Cicero**

answered Mar 21 '13 at 12:44



Michael Tiller

5,623 ● 3 ● 13 ● 27

... Volker Waurich:

- „Wie wäre es mit einer externen Funktion, die Speicher bereithält?“
  - **ExternalMemoryLib**  
(<https://github.com/vwaurich/ExternalMemoryLib>)
    - Funktion schreibt Daten in einen Speicher und kann zu jedem beliebigen Zeitpunkt wieder aufgerufen werden.

Aufbau:

- Externe Objekte mit constructor und destructor
- Externe C Funktionen

Funktionen:

- Set und get von Integer, Real, Boolean, Real range
- Set und get von Real, Real range mit Zeitkontrolle

## Anwendung auf Minimalbeispiel:

model Minimum

```
import EM = BFT.ExternalMemoryLib;
import Modelica.Constants.pi;
```

```
Real x(start=3, fixed=true);
Real derx(start=-1, fixed=true);
Real y1;
```

```
EM.ExternalMemoryReal globalMin =
  EM.ExternalMemoryReal(1);
```

algorithm

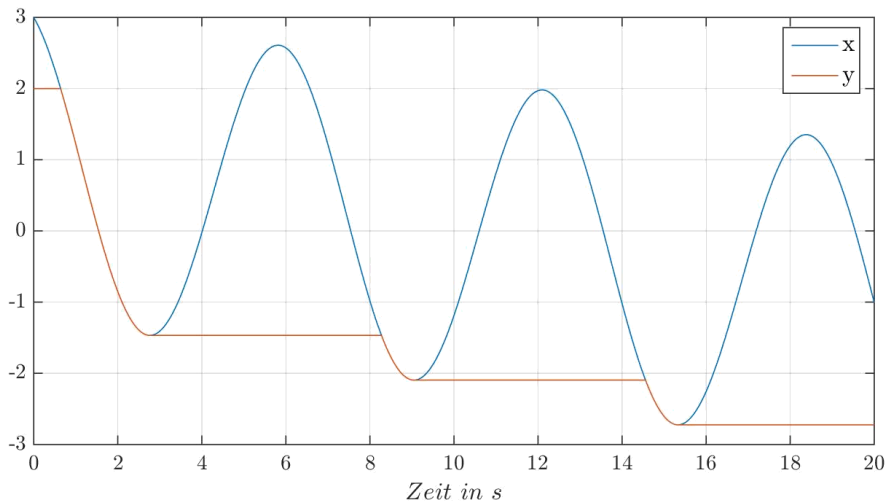
```
y1 := min(EM.Functions.getReal(globalMin, 1,time), x);
EM.Functions.setReal(globalMin,1,y1,time);
```

initial equation

```
EM.Functions.setReal (globalMin,1,2,time);
```

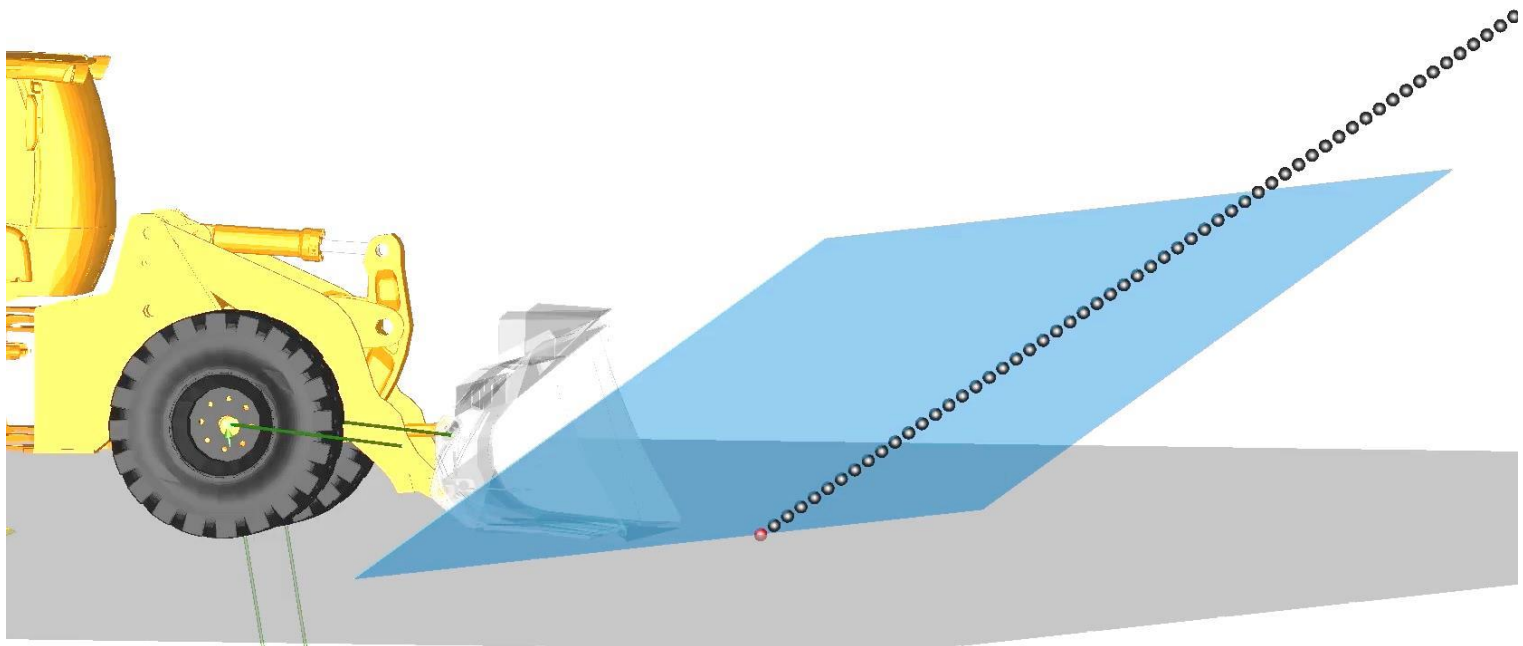
equation

```
der(x) = derx;
der(derx) = -x+1-0.1*time;
end Minimum;
```



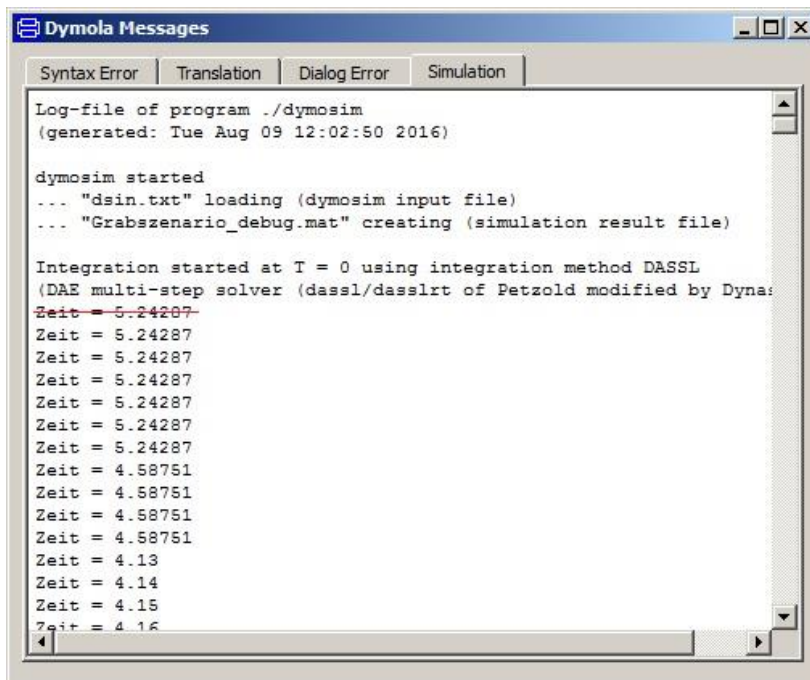
Minimalbsp. funktioniert! → Anwendung auf das Haufwerk!

- „Springende“ Punkte
- Bei Funktionsaufrufen eines ungültigen Schrittes wird Variable dennoch gesetzt. (debugging: `Modelica.Utilities.Streams.print`)
- Es dürfen nur gültige Schritte verwendet werden!



## Minimalbsp. funktioniert! → Anwendung auf das Haufwerk!

- „Springende“ Punkte
- Bei Funktionsaufrufen eines ungültigen Schrittes wird Variable dennoch gesetzt. (debugging: Modelica.Utilities.Streams.print)
- Es dürfen nur gültige Schritte verwendet werden!

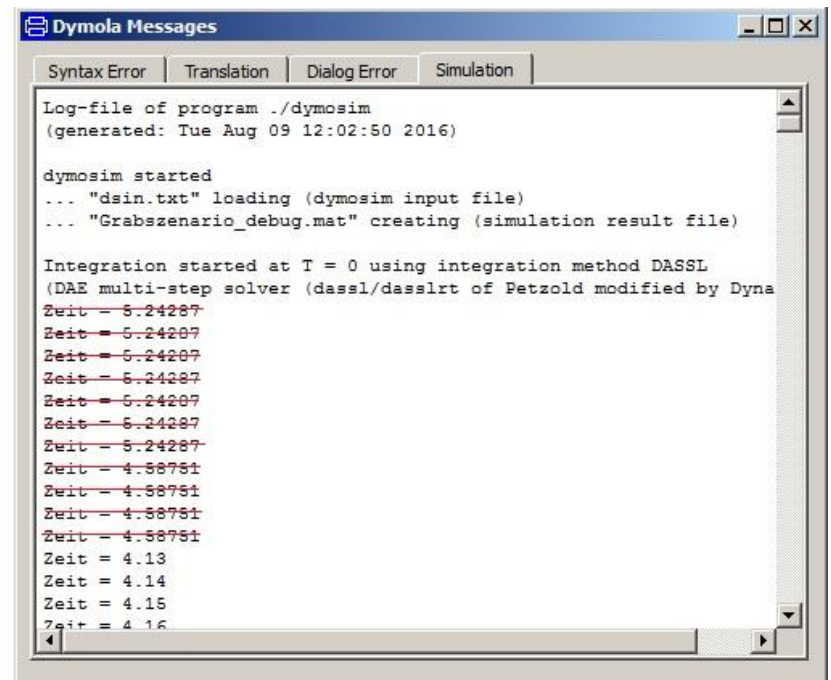


```

Dymola Messages
Syntax Error | Translation | Dialog Error | Simulation
Log-file of program ./dymosim
(generated: Tue Aug 09 12:02:50 2016)

dymosim started
... "dsin.txt" loading (dymosim input file)
... "Grabszenario_debug.mat" creating (simulation result file)

Integration started at T = 0 using integration method DASSL
(DAE multi-step solver (dassl/dasslrt of Petzold modified by Dyna:
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 4.58751
Zeit = 4.58751
Zeit = 4.58751
Zeit = 4.58751
Zeit = 4.13
Zeit = 4.14
Zeit = 4.15
Zeit = 4.16
  
```



```

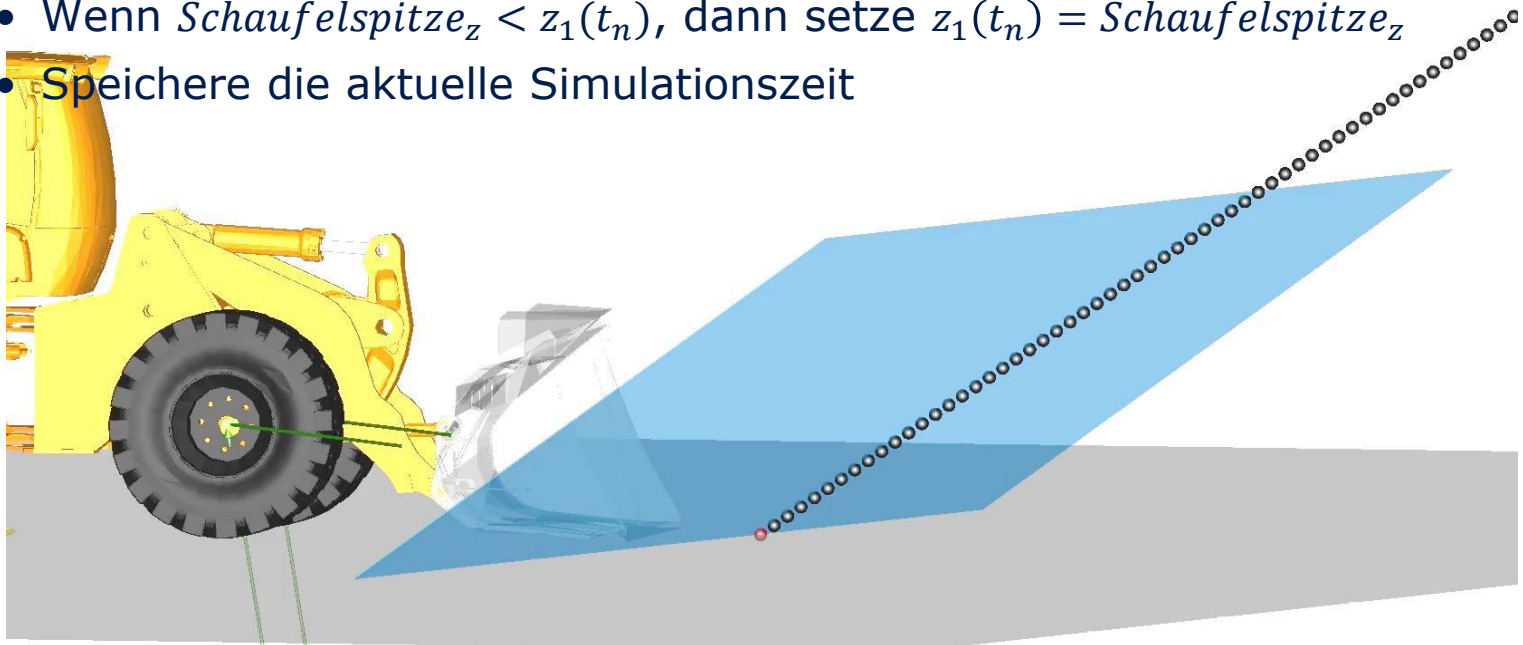
Dymola Messages
Syntax Error | Translation | Dialog Error | Simulation
Log-file of program ./dymosim
(generated: Tue Aug 09 12:02:50 2016)

dymosim started
... "dsin.txt" loading (dymosim input file)
... "Grabszenario_debug.mat" creating (simulation result file)

Integration started at T = 0 using integration method DASSL
(DAE multi-step solver (dassl/dasslrt of Petzold modified by Dyna
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 5.24287
Zeit = 4.58751
Zeit = 4.58751
Zeit = 4.58751
Zeit = 4.58751
Zeit = 4.13
Zeit = 4.14
Zeit = 4.15
Zeit = 4.16
  
```

## Korrektur der fehlerhaften Zeitschritte (Bsp.)

- Annahme, wenn  $t_n > t_{n-1}$  folgt, dass  $t_{n-1}$  ein gültiger Schritt ist.
- Ablauf:
  - Wenn  $t_n > t_{n-1}$ , dann setze die Originalwerte ( $z_0(t_n)$ ) auf  $z_1(t_{n-1})$
  - Zurücksetzen aller Modifikationen von  $z_1(t_n)$  auf  $z_0(t_n)$
  - Wenn  $Schaufelspitze_z < z_1(t_n)$ , dann setze  $z_1(t_n) = \text{Schaufelspitze}_z$
  - Speichere die aktuelle Simulationszeit



Algorithmus wird anders interpretiert als in Dymola

➤ Bodenkraft = 0.

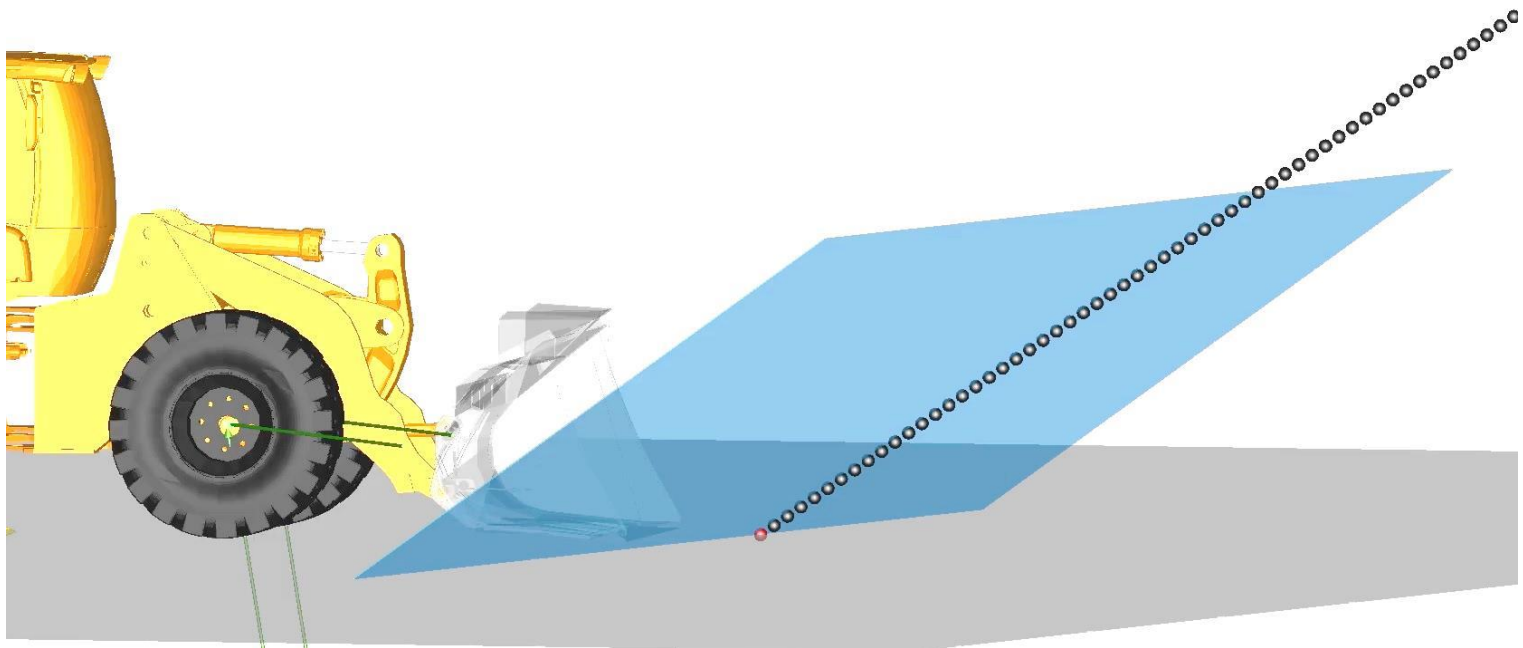
Zusätzliche Möglichkeiten mit SimX

- `analysisTypeDetail() == "validStep,,`
  - Erspart den Workaround mit dem gültigen Schritt
- `Last()` Operator
  - Ersetzt die `ExternalMemoryLib`

Problem: Nicht Modelica konform

## Finales Bodenmodell

- Gesamter Algorithmus als externe C Funktion





	<b>Nicht- lineares GS</b>	<b>Rechenzeit [s]</b>	<b>Sim Zeit [s]</b>	<b>Korrektes Ergebnis</b>
Ohne Kopplung	57	2,5	8	Nein
Mit Kopplung	57	4,5	8	Nein
When sample	0	36	8	Ja
ExtMem	0	1,67	8	Ja
Finales Modell	0	1,67	8	Ja

## Nützliches Feature:

- C-Code direkt in den Modelica Tools schreiben zu können
- Funktionsrümpfe (Konstruktor, Destruktor) automatisch erstellen
- Oder: Kopplung mit VisualStudio



**»Wissen schafft Brücken.«**

```
if noEvent(time>getReal(Zeitstempel, 1)) then
    // Der letzte Zeitschritt ist gültig!
    // Die Modifikation vom letzten Zeitschritt ist der neue Ausgangspunkt
    setRealRange(Hauf_mod_0,1,size(Hauf,2),getRealRange(Hauf_mod,1,size(Hauf,2)));
end if;

// Alle Modifikationen werden zurückgesetzt, falls mehrere ungültige Schritte auftraten.
    setRealRangeAt(Hauf_mod,1,size(Hauf,2), getRealRangeAt(Hauf_mod_0,1,size(Hauf,2)));

// Wenn Schaufelspitze im Haufwerk, wird die Oberfläche bis zur Schaufelspitze
// verschoben.
if noEvent(r_SS[3] < getRealValueAt(Hauf_mod_0, ii)) then
    setRealValueAt(Hauf_mod,ii,r_SS[3]);
end if;

// Hauf_mod ist ein Record. Zum Arbeiten wird aber eine Realvariable benötigt.
    Haufwerk_z := getRealRangeAt(Hauf_mod, 1, size(Hauf,2));

// Der aktuelle Zeitschritt wird zwischengespeichert
    setRealValueAt(Zeitstempel,1,time);
```