10th UCAAT User Conference on Advanced Automated Testing

ETSI The Standards People

Mutation Testing in Additive Manufacturing

Dr. Johannes Erbel

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

16/11/2023

Content

1) Additive Manufacturing
2) Mutation Testing in Additive Manufacturing
3) Feasibility Study: Gcode Mutation Operators
4) Lessons Learned
5) Conclusion

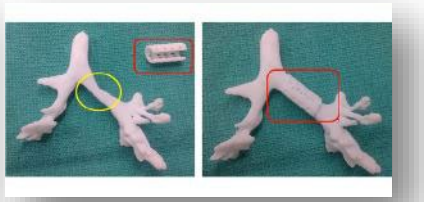
Additive Manufacturing (AM)

Foundations of 3D printing

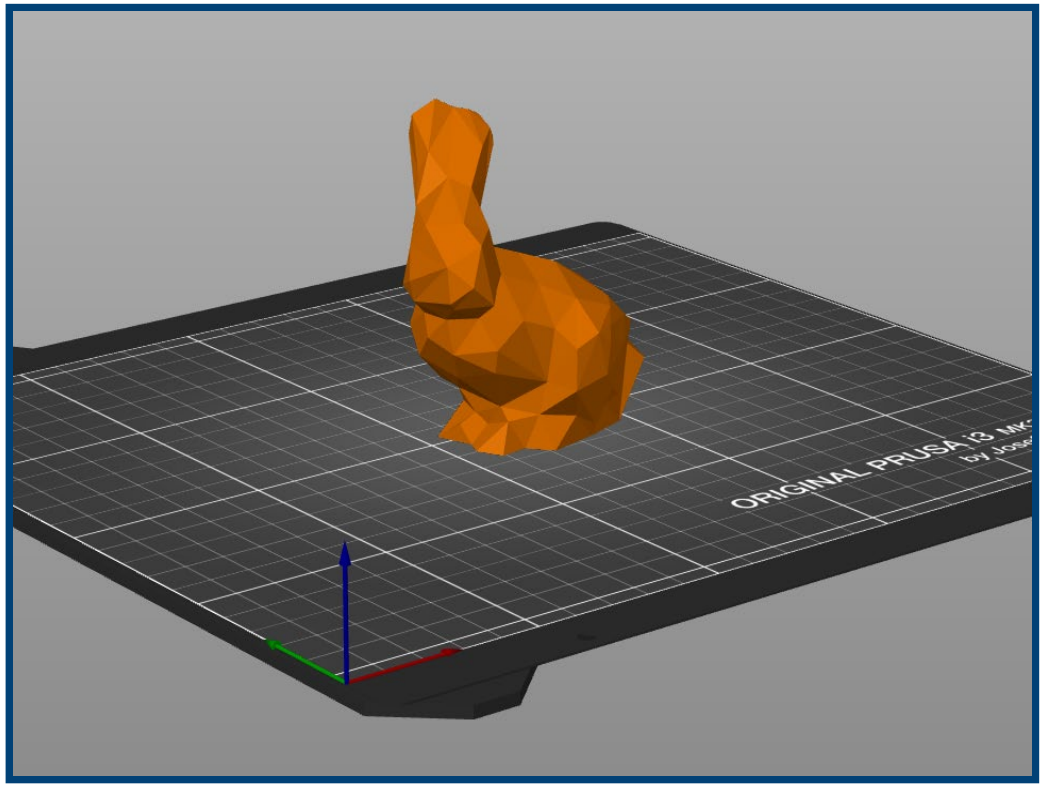
Layer-wise Fabrication

Hasbro, Toy Puzzle

AM Aircraft Hinge with
50% less weight (EADS)



Custom airway stent
(U. Michigan)

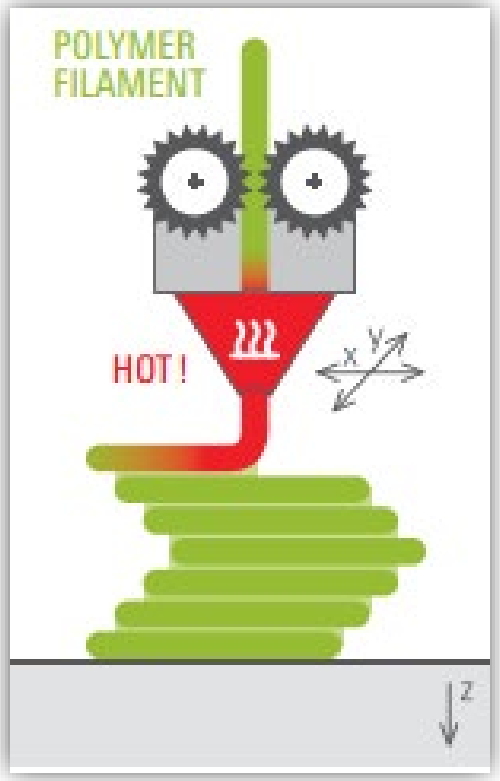


PrusaSlicer

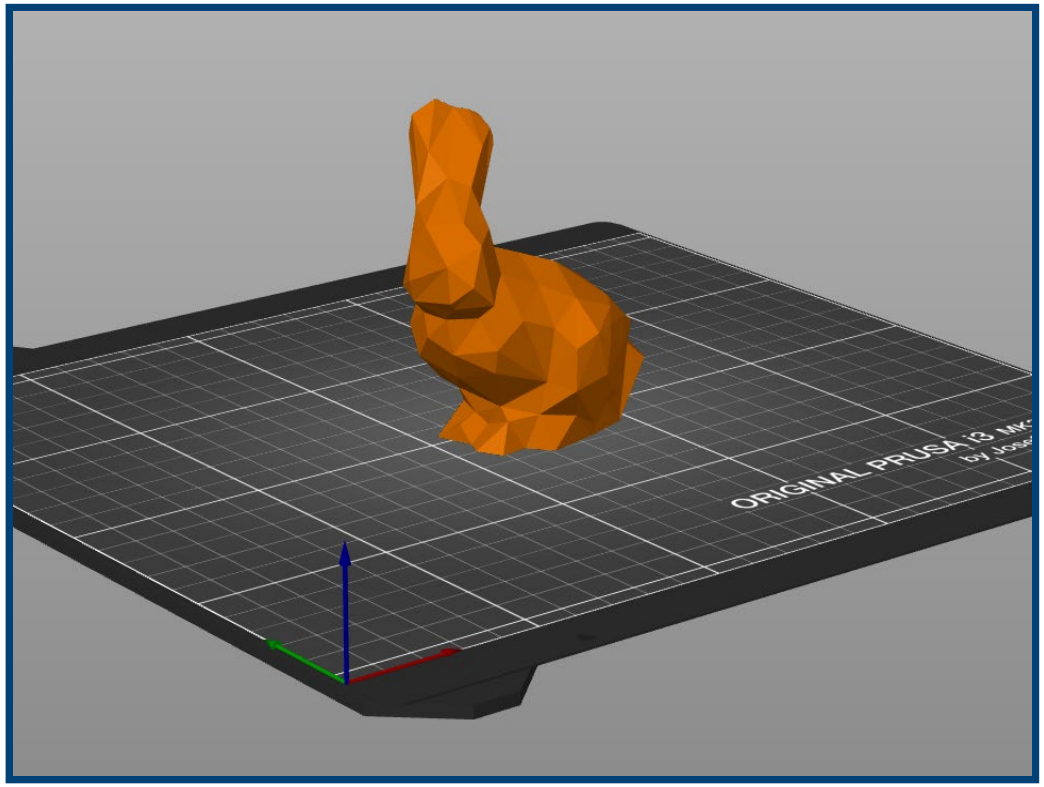
Additive Manufacturing (AM)

Foundations of 3D printing

Layer-wise Fabrication



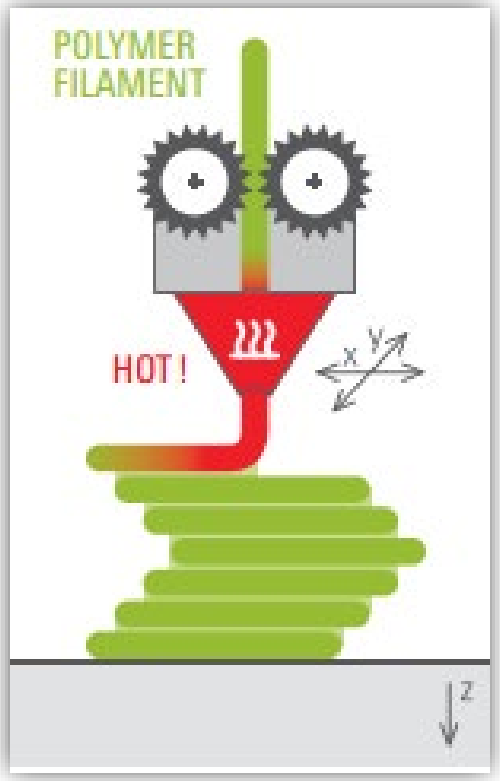
Formnext, AM Field Guide, Messe Frankfurt, 2021



Additive Manufacturing (AM)

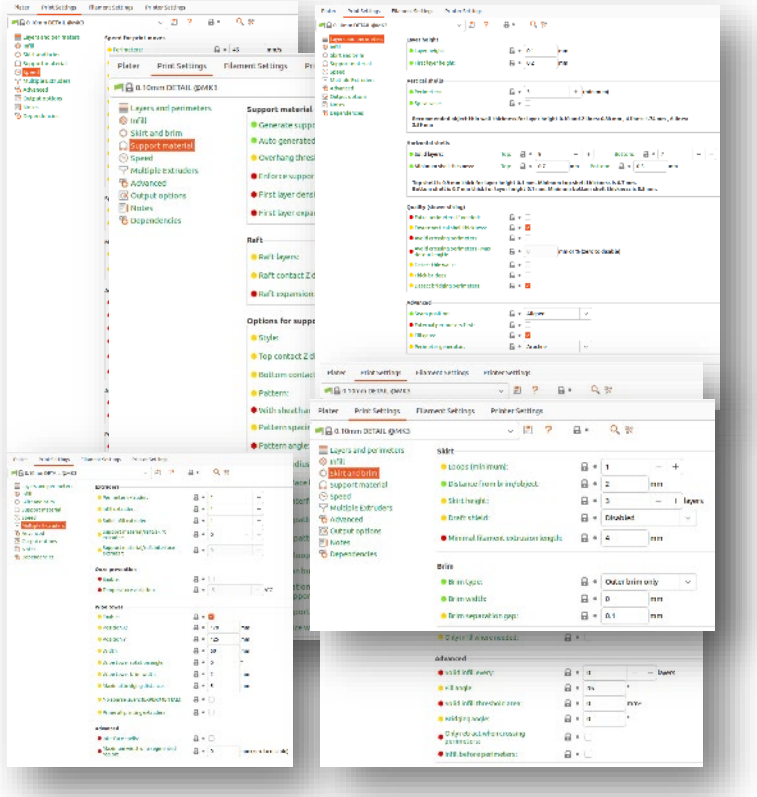
Difficulties in AM

Layer-wise Fabrication



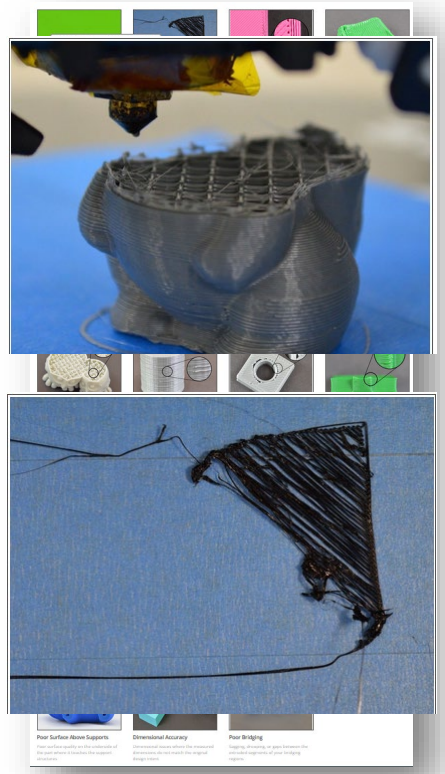
Formnext, AM Field Guide, Messe Frankfurt, 2021

Slicing and Instructions



PrusaSlicer

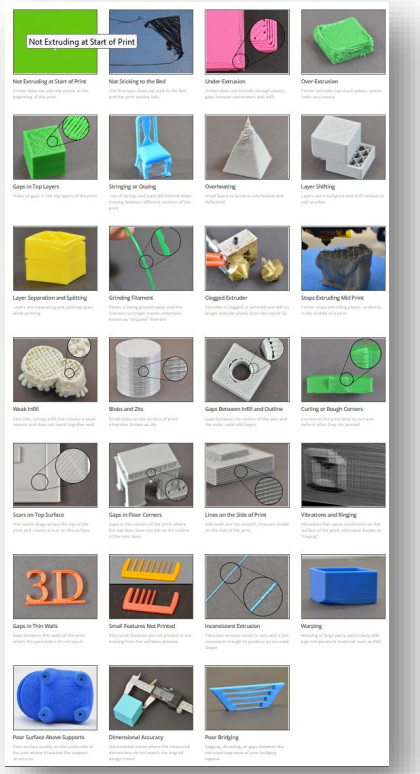
Huge Defect Variety



Simplify3D

Additive Manufacturing (AM)

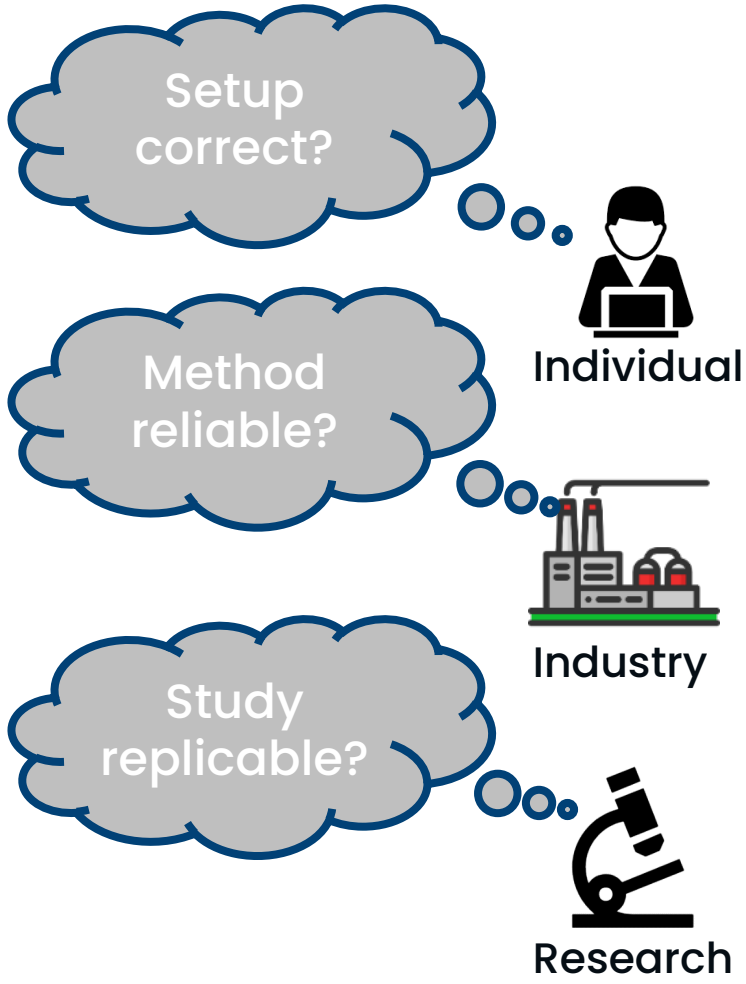
Huge Defect Variety



Simplify3D

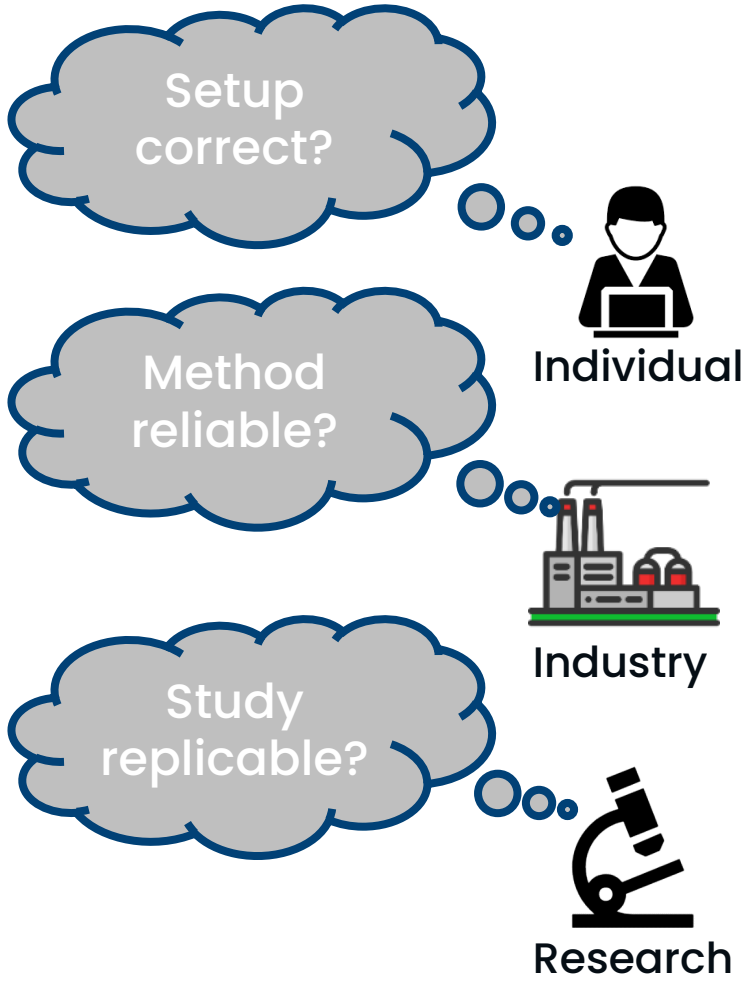
Defect Detection Methods

Obico

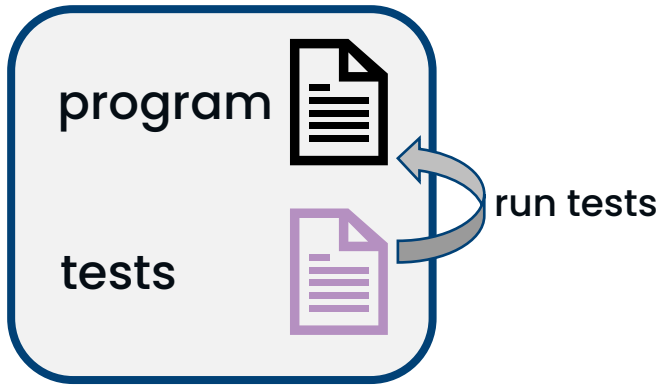


Content

1) Additive Manufacturing
2) Mutation Testing in Additive Manufacturing
3) Feasibility Study: Gcode Mutation Operators
4) Lessons Learned
5) Conclusion



Mutation Testing in Software Engineering

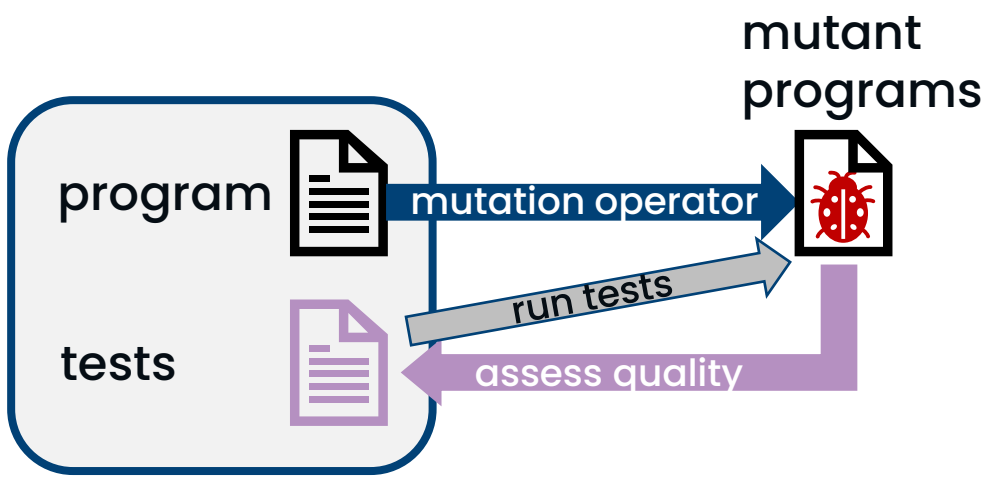



```python
def bubbleSort(arr):
    n = len(arr)
    # optimize code, so if the array is already sorted, it doesn't need
    # to go through the entire process
    swapped = False
    # Traverse through all array elements
    for i in range(n-1):
        # range(n) also work but outer loop will
        # repeat one time more than needed.
        # Last i elements are already in place
        for j in range(0, n-i-1):

            # traverse the array from 0 to n-i-1
            # Swap if the element found is greater
            # than the next element
            if arr[j] > arr[j + 1]:
                swapped = True
                arr[j], arr[j + 1] = arr[j + 1], arr[j]

        if not swapped:
            # if we haven't needed to make a single swap, we
            # can just exit the main loop.
            return
```

program

Bubblesort, geeksforgeeks

Mutation Testing in Software Engineering



To what extent can my tests find bugs in my source code?

Adapted from Petrovic et al., 2021

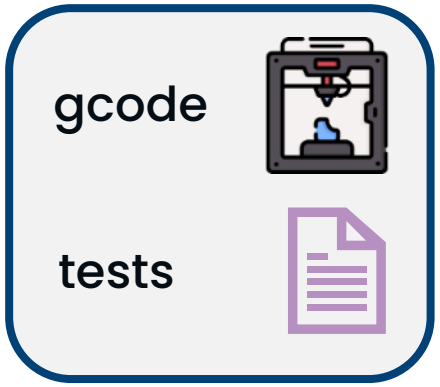
```
def bubbleSort(arr):
    n = len(arr)
    # optimize code, so if the array is already sorted, it doesn't need
    # to go through the entire process
    swapped = False
    # Traverse through all array elements
    for i in range(n+1):
        # range(n) also work but outer loop will
        # repeat one time more than needed.
        # Last i elements are already in place
        for j in range(0, n-i-1):

            # traverse the array from 0 to n-i-1
            # Swap if the element found is greater
            # than the next element
            if arr[j] > arr[j + 1]:
                swapped = True
                arr[j], arr[j + 1] = arr[j + 1], arr[j]

        if not swapped:
            # if we haven't needed to make a
            # can just exit the main loop.
            return
```

mutation operator

mutant program

Bubblesort, geeksforgeeks

Mutation Testing in AM



- One Instruction per Line
  - Comments
  - Movement
    - Feedrate (Speed)
    - Coordinates
    - Filament Extrusion
  - Misc
    - Temperature
    - Fan
    - Progress

```
;TYPE:External perimeter
G1 F1200
G1 X81.41 Y81.41 E.53867
G1 X98.59 Y81.41 E.53867
G1 X98.59 Y98.59 E.53867
G1 X81.47 Y98.59 E.53678
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E-.912
;WIPE_END
G1 E-.048 F2700
G1 Z.4 F720
```

object perimeter

gcode

3D printer images from: www.freepik.com

Mutation Testing in AM

gcode 

tests 

- One Instruction per Line
  - Comments
  - Movement
    - Feedrate (Speed)
    - Coordinates
    - Filament Extrusion
  - Misc
    - Temperature
    - Fan
    - Progress

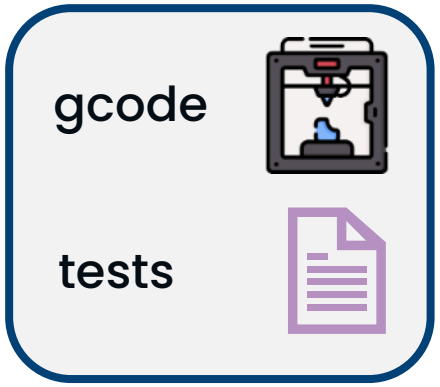
```
;TYPE:External perimeter
G1 F1200
G1 X81.41 Y81.41 E.53867
G1 X98.59 Y81.41 E.53867
G1 X98.59 Y98.59 E.53867
G1 X81.47 Y98.59 E.53678
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E-.912
;WIPE_END
G1 E-.048 F2700
G1 Z.4 F720
```

gcode

object perimeter

1200 mm/s

Mutation Testing in AM



- One Instruction per Line
  - Comments
  - Movement
    - Feedrate (Speed)
    - Coordinates
    - Filament Extrusion
  - Misc
    - Temperature
    - Fan
    - Progress

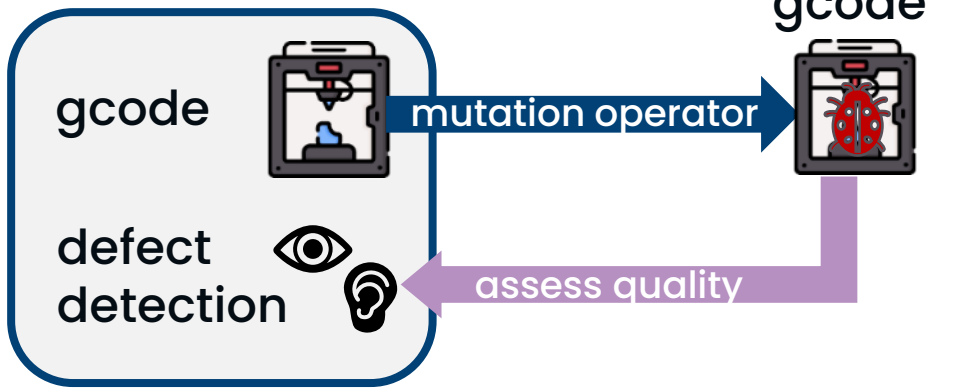
```
;TYPE:External perimeter
G1 F1200
G1 X81.41 Y81.41 E.53867
G1 X98.59 Y81.41 E.53867
G1 X98.59 Y98.59 E.53867
G1 X81.47 Y98.59 E.53678
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E-.912
;WIPE_END
G1 E-.048 F2700
G1 Z.4 F720
```

object perimeter

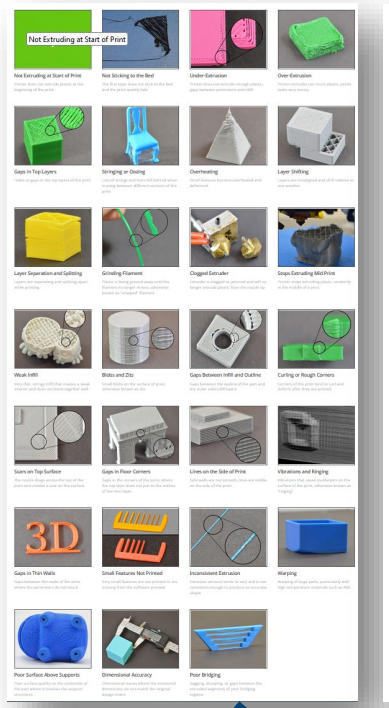
1200 mm/s

gcode

Mutation Testing in AM



To what extent can mutation operators imitate print defects to evaluate defect detection methods?



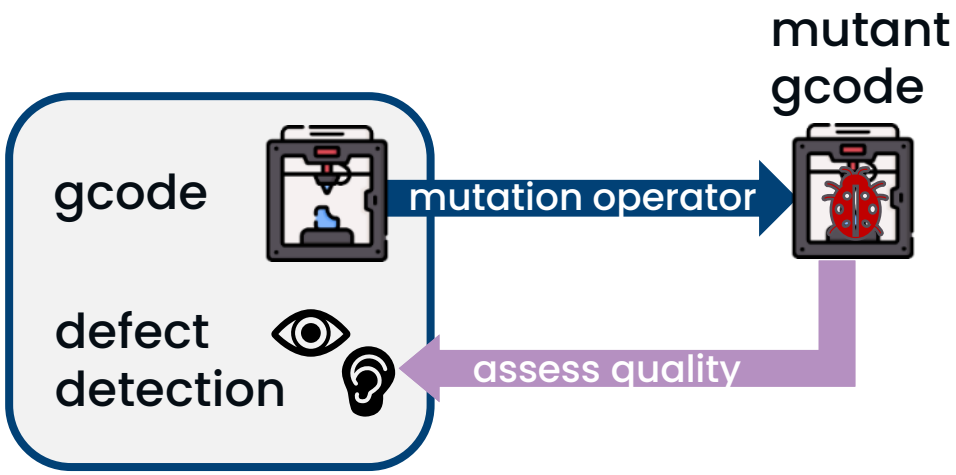

```
;TYPE:External perimeter
G1 F1200
G1 X81.41 Y81.41 E.53867
G1 X98.59 Y81.41 E.53867
G1 X98.59 Y98.59 E.53867
G1 X81.47 Y98.59 E.53678
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E-.912
;WIPE_END
G1 E-.048 F2700
G1 Z.4 F720
```

object perimeter

1200 mm/s

gcode

Mutation Testing in AM



Control severity

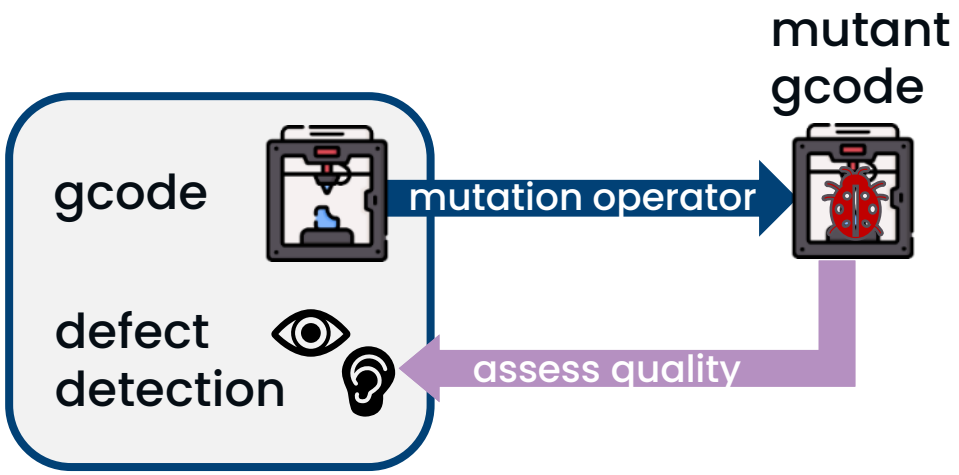
- Instruction Wise

```
;TYPE:External perimeter
G1 F10000
G1 X81     Y81.41 E.53867
G1 X98.59 Y81     E.53867
G1 X98.59 Y98.59 E0
G1 X81.47 Y98.59 E.53678
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E-.912
;WIPE_END
G1 E-.048 F2700
G1 Z.4 F720
```

mutation operator

mutant gcode

Mutation Testing in AM



Control severity

- Instruction Wise
- Feature Wise

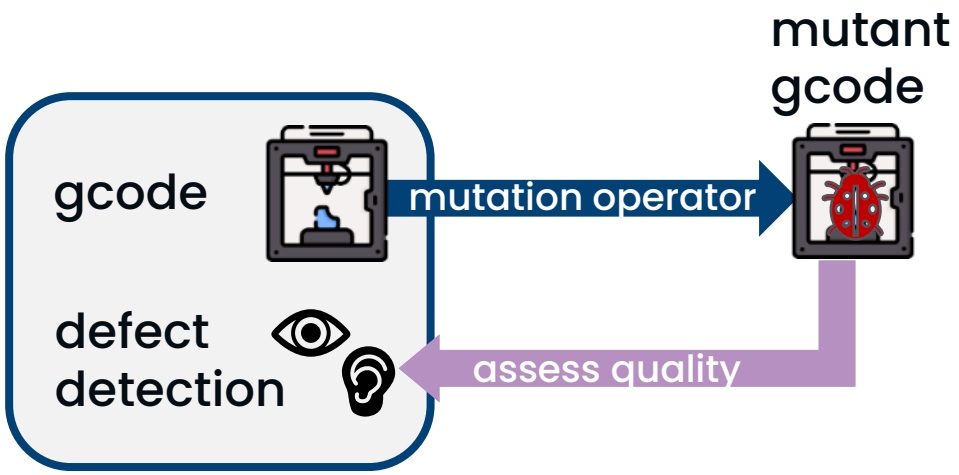
```
;TYPE:External perimeter
G1 F1200
G1 X81.41 Y81.41 E2
G1 X98.59 Y81.41 E0.3
G1 X98.59 Y98.59 E0.3
G1 X81.47 Y98.59 E2
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E-.912
;WIPE_END
G1 E-.048 F2700
G1 Z.4 F720
```

mutation operator

1200 mm/s

mutant gcode

Mutation Testing in AM



Control severity

- Instruction Wise
- Feature Wise
- Layer Wise

```
;TYPE:External perimeter
G1 F1200
G1 X81.41 Y81.41 E0
G1 X98.59 Y81.41 E0
G1 X98.59 Y98.59 E0
G1 X81.47 Y98.59 E0
M204 P1000
G1 X81.514 Y98.204 F9000
G1 E-2.24 F2700
;WIPE_START
G1 F7200
G1 X81.461 Y96.03 E0
;WIPE_END
G1 E0      F2700
G1 Z.4 F720
```

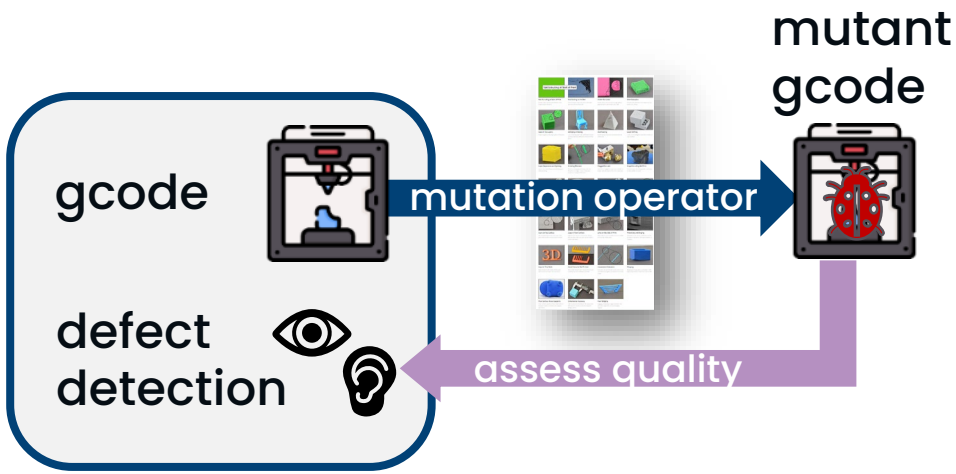
mutation operator

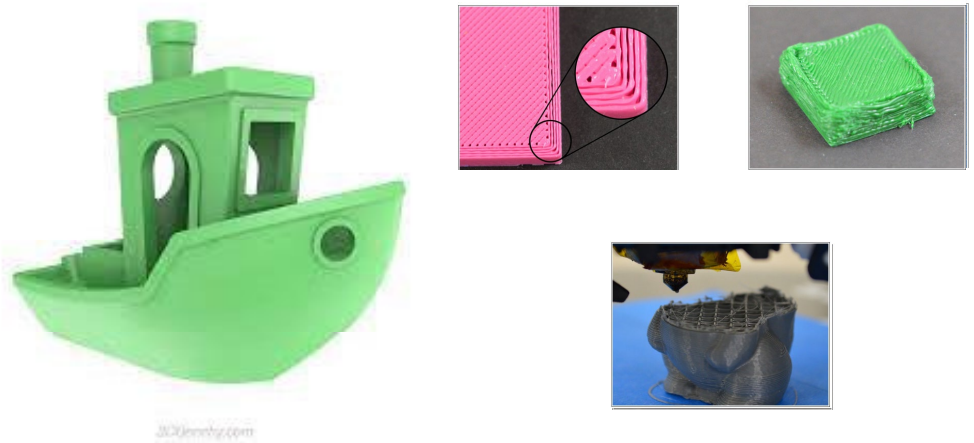
1200 mm/s

mutant gcode

Content

1) Additive Manufacturing
2) Mutation Testing in Additive Manufacturing
3) Feasibility Study: Gcode Mutation Operators
4) Lessons Learned
5) Conclusion





Under Extrusion

Under Extrusion

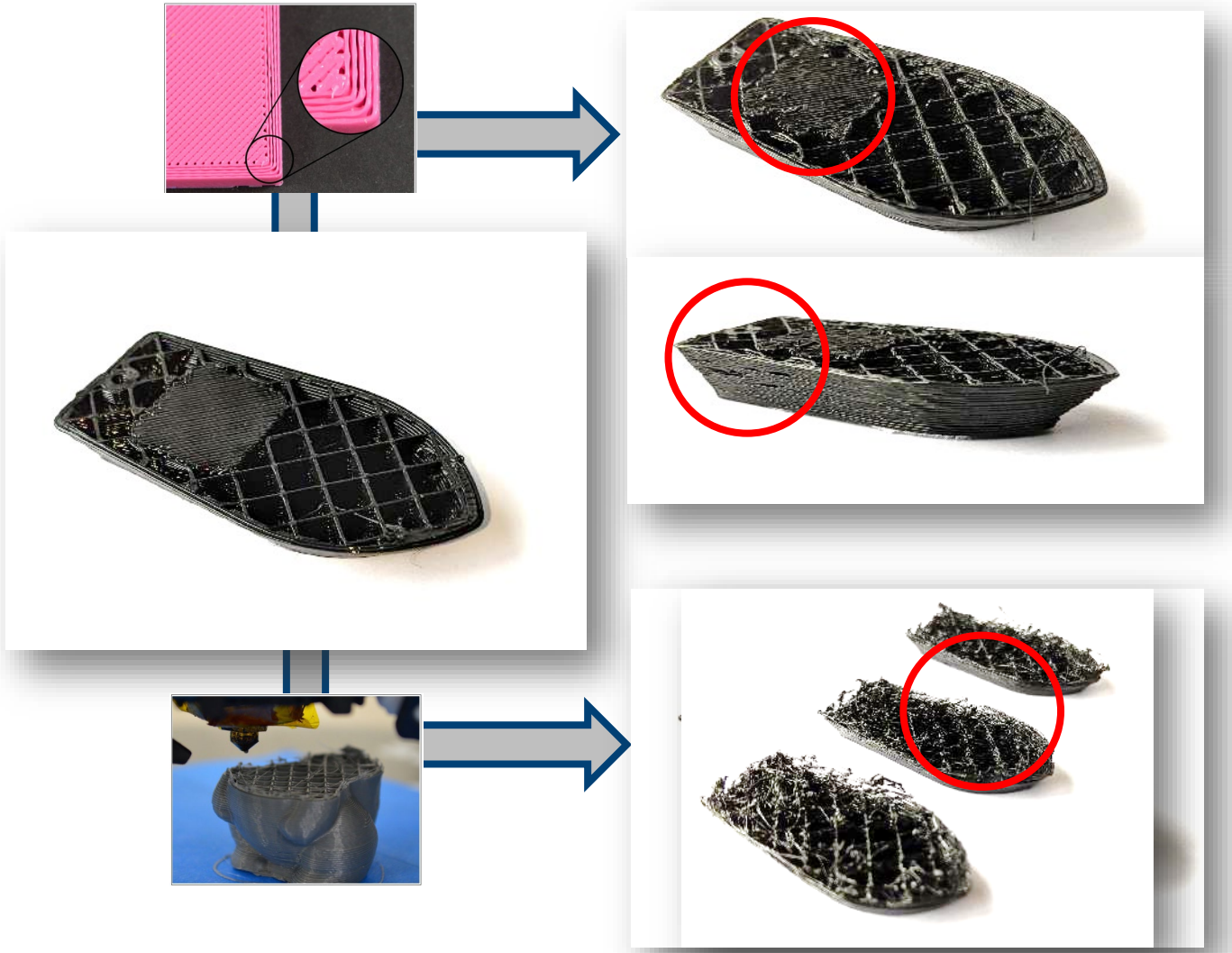
- Printer extrudes less plastic

Reason

- Clogged Nozzle
- Wrong slicer configuration

Approach

- Mutate extrusion parameter
  1. Constant layerwise reduction
  2. Increasing reduction layerwise

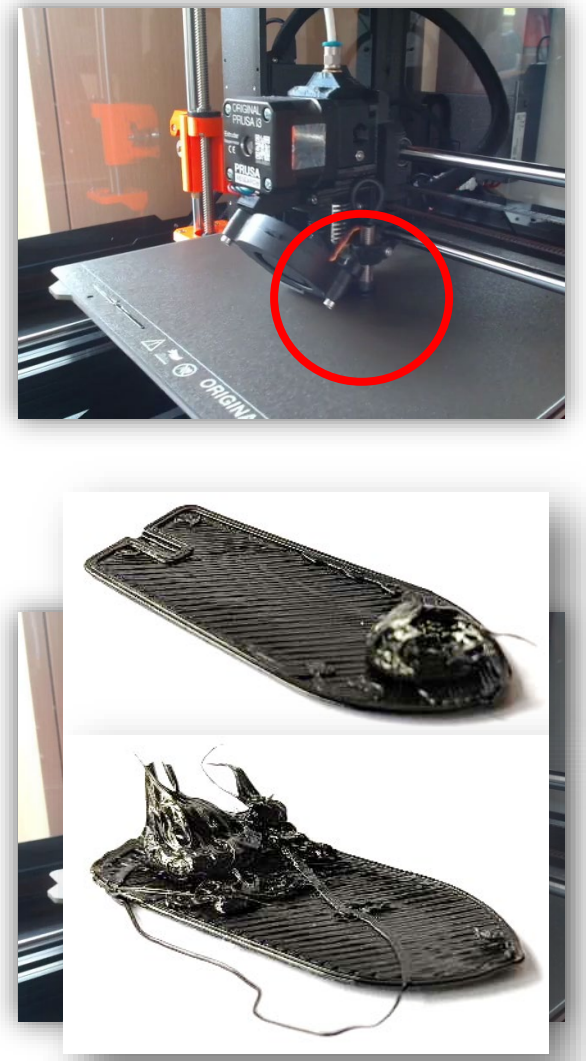


Not Sticking to Bed

Not Sticking to Bed
- Print does not stick
- Quickly fails

Reason
- Dirty print bed
- Speed to high
- Temperature issues
- Nozzle to high

Approach
- Mutate speed, temperature, height
- Start with layer 2



Content

1) Additive Manufacturing
2) Mutation Testing in Additive Manufacturing
3) Feasibility Study: Gcode Mutation Operators
4) Lessons Learned
5) Conclusion

Lessons Learned

- Real problems reproducible by gcode mutants
  - Depending on operator near same results
- Mutations are time consuming
  - First-layer quicker than mid-layer
- First-layer rather severe defects
  - Leave first two layers intact
- Defect masking
  - Small deviations quickly lead to larger faults

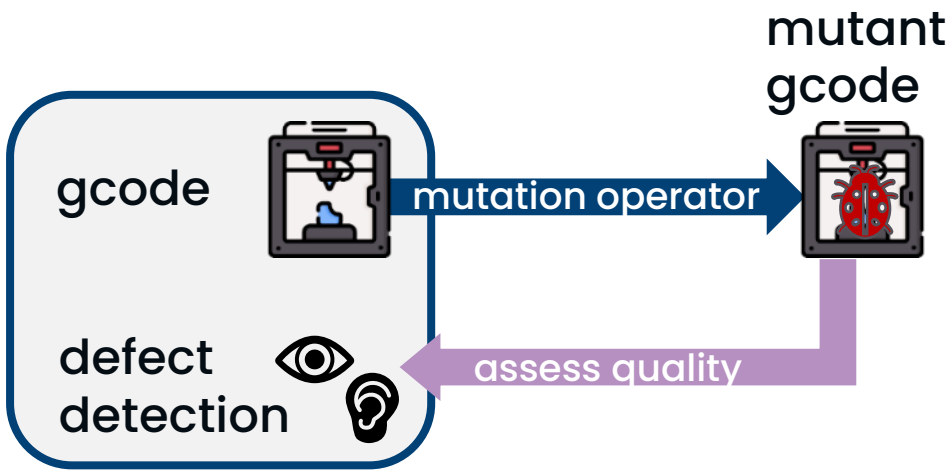
Conclusion

Mutation operators

- Can be applied to gcode
- Can partially imitate print defects
- May serve as benchmark for defect detection
- Non intrusive and repeatable

Future work

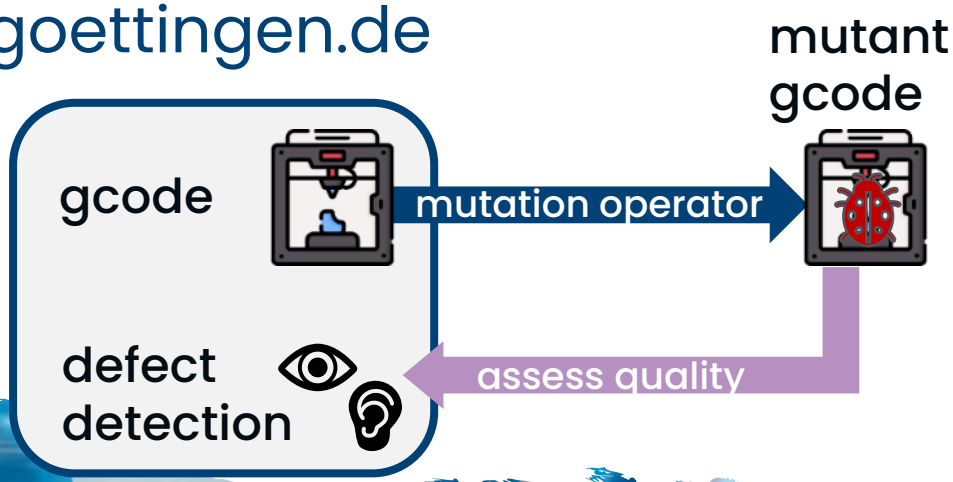
1. Quantitative evaluation
   - Different prints
   - Different printers
   - Comprehensive mutant mapping
2. Further analysis
   - Mutant injection points
   - Mutant dependencies



To what extent can mutation operators imitate print defects to evaluate defect detection methods?

Thank you for your attention!

Contact me:
johannes.erbel@cs.uni-goettingen.de



Backup

Warping

Warping
- Curled corners
- High temperature prints

Reason
- Plastic shrinks when cooled

Approach
- Layer injection
- Repeat perimeter movement
- Turn on fan
- Turn heatbed off