

2D Bauteilverschachtelung / Plating / Nesting / Bin Packing

Allgemeines zum Nesting

Verschnittoptimierung, Zuschnittoptimierung, cutting plan optimizer, Formschachtelung, Nesting, Verschachtelung von Konturen oder Bauteilen, Schachteln, Plating, Bin Packing, scrap minimizing - es gibt viele Namen und Stichwörter für ein- und dasselbe Grundproblem. Nesting ist ein Prozess, der eine wichtige Rolle in der Fertigung trägt, denn er ist für die effiziente und materialsparsparende Herstellung, sowie für eine intelligente bzw. platzsparende Lagerung bzw. Transport von Zuschnitten äußerst relevant. Typischerweise häufig benötigt werden Nesting-Programme beim Schneiden von Werkstoffen wie Holzplatten, Stoffen, Leder, Metall oder Glas.

Im Vergleich zu einer manuellen oder keiner Verschachtelung ist das Berechnen einer guten Nesting-Lösung für ein geplantes Projektvorhaben in der Regel zu empfehlen - insbesondere wenn viele Teile benötigt werden. Der Initialaufwand ist viel höher, jedoch lassen sich langfristig Material, Zeit und Aufwand im beträchtlichen Ausmaß einsparen.

Optimierung als Ziel

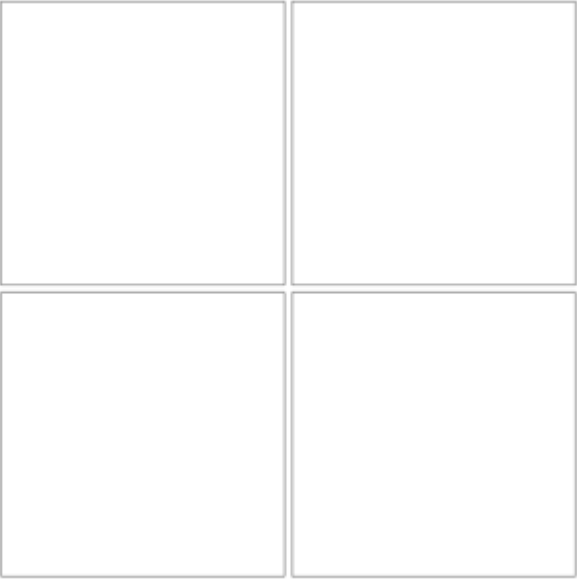
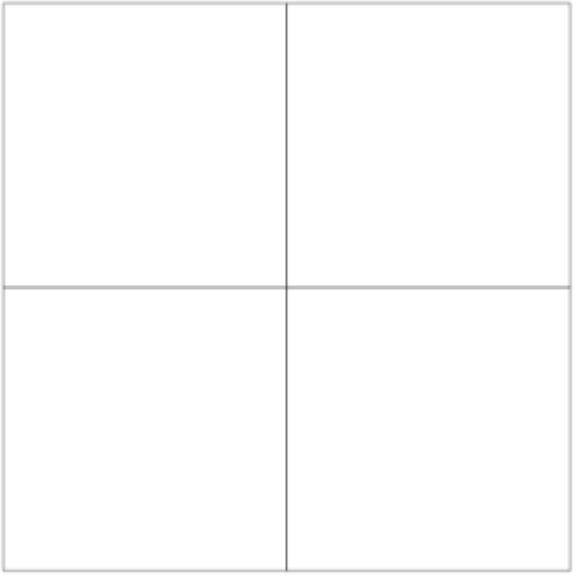
Typische Ziele vom Nesting sind zum Beispiel:

- Teile eng an eng platzieren
- Plattengrößen reduzieren
- Abfall bzw. Ausschuss reduzieren
- Schneidzeit optimieren / Zeit sparen
- Lagerplatzbedarf verkleinern
- Rüstzeiten / Materialwechselzeiten verkürzen
- ...

Viele dieser genannten Punkte bedingen sich dabei gegenseitig.

Reduktion von Schnittkanten und Schnittzeiten, höhere Materialausbeute

Beim Nesting werden die zu schneidenden Teile so angeordnet, dass die Materialmenge maximiert wird. In vielen Fällen ermöglicht es das Schneiden auf einer gemeinsamen Linie. Die auf der Platte angeordneten Elemente teilen sich gemeinsame Schnittlinien, indem die Teile so angeordnet werden, dass sie eine gemeinsame Kante haben. Nehmen wir zum Beispiel an, es sollen vier einzelne Quadrate geschnitten werden. Anstatt sie einzeln zu schneiden und damit 16 einzelne Schnitte zu fahren, müssen demnach stattdessen nur 6 Schnitte abgefahren werden.

4 einzelne Quadrate	Nesting mit geteilten Schnittlinien
	

In diesem Beispiel hat sich durch die Verschachtelung die Anzahl der Schnitte um 63 % verringert (wobei der Schneidweg selbst nicht um 63 % gesunken ist).

Die Verschachtelung ermöglicht eine höhere Ausbeute des Ausgangsmaterials. Es ist nicht unüblich, die Produktausbeute um 40 % zu erhöhen und gleichzeitig die Schnittzeit zu verringern. Das Verschachteln kann eine enorme Zeit- und Materialersparnis bedeuten, wenn es richtig und sorgfältig durchgeführt wird. Wenn jedoch eine Schnittlinie versagt, verliert man unter Umständen nicht nur ein Teil, sondern gleich zwei. Nach dem Nesting muss also sorgfältig geprüft werden, ob alles stimmt.

Auch übrige Restmaterialplatten können noch genutzt werden

Mit Hilfe von Nestingprogrammen lassen sich unter anderem auch Restplatten besser ausnutzen. Einige Nestingprogramme unterstützen das Verschachteln von Bauteilen auf ungewöhnlich geformten (nicht-triviale Rechtecke) Plattenkonturen bzw. Platten, die Löcher und Aussparungen enthalten. Die Konturen von Restplatten können beispielsweise

fotografiert und entzerrt oder mit einem Scanner gescannt werden. Auf diese Weise lassen sich insbesondere kostenintensive Reste, die häufig als Abfall gelten, noch verwerten.

Zwischenabstände, Stege und Lagerung

Neben der Optimierung der Schnittzeit durch Reduzieren auf gemeinsame Bauteilkanten nimmt das Nesting noch eine andere wichtige Rolle ein. Statt sich die Bauteilkanten zu teilen können die einzelnen Teile auf einer Platte auch mit geringfügigen Zwischenabständen angeordnet werden. Das erlaubt das Einfügen von kleinsten Stegen bzw. Brücken (sogenannten "tabs"), die die Teile an ihrer Schnittkontur minimal festhalten. Das verhindert, dass die Teile nach dem Schneiden einfach aus ihrem Negativ herausfallen. Hierbei wird die Transportierbarkeit der Platten verbessert und ein Plattenwechsel in der Maschine wird beschleunigt. Außerdem lassen sich diese Platten u.U. gut an den Kunden weitertransportieren oder gut lagern. Das Einfügen von Stegen funktioniert nur, wenn zwischen den geschachtelten Bauteilen genügend Platz ist (Abfallmaterial). Anderfalls würde die Geometrie der benachbarten Teile ungewollt durch zusätzliche Linien bzw. Unterbrechungen modifiziert werden. Stege verringern die Kantenqualität, da beim Heraustrennen feine Bruchkanten entstehen, welche gegebenenfalls überschliffen werden müssen. Weitere Informationen zu Stegen finden sich auf der Seite [Ein- und Auslaufschnitte, Stege, Brücken, Laschen](#).

Nesting gibt es nicht nur in der zweidimensionalen Ebene. Auch in Volumina kann geschachtelt werden. Siehe auch [3D Bauteilverschachtelung / Plating / Nesting / Bin Packing](#)

Aufwand und allgemeine Probleme beim Nesting von 2D-Bauteilen

Bezogen auf die große Auswahl an Software ist es leider so, dass viele Programme nicht mit allen Schwierigkeiten umgehen, die beim Verschachteln auftreten können. Sie unterscheiden sich stark im Funktionsumfang, der Qualität des Nestings, der Fehleranfälligkeit, der Plattformkompatibilität, den möglichen Input-Formaten wie SVG oder DXF. Häufig müssen Geometrien hierbei erst von einem Format in ein anderes Format konvertiert werden, was zu Aufwand und Fehlern führen kann (z.B. falsche Skalierung durch mangelnde Einheitensystemkompatibilität, fehlende Linien oder Layer, Konvertierung von Splines zu Polylinien, unstimmmige Liniendicken oder -typen, etc.). Der Aufwand des Nestings steigt mit den Anforderungen. Je mehr (Rest)platten und je mehr zu

schachtelnde Teile anfallen, desto komplexer ist die Berechnung einer idealen Verschachtelung. Wenn die Teile Texturen oder andere Arrangements enthalten, die beim Verschachteln nicht verloren gehen dürfen, steigt der Aufwand ebenso. Auch im Nachgang bedarf es häufig einer weiteren Optimierung, zum Beispiel der Sortierung der Reihenfolge der Linien, um kurze Schneid- und Reisewege (travel moves) für das spätere Werkzeug (z.B. Laser, Schneidmesser, Fräser) zu ermöglichen.

Wichtig beim Verschachteln, aber häufig nicht beachtet, sind zum Beispiel

- Möglichkeit, Teile abstandslos zu platzieren. In diesem Fall teilen sich geschachtelte Teile gemeinsame Kanten. Hierbei werden genau genommen Bauteilkanten entfernt. Werden diese Kanten nicht entfernt, dann bearbeitet das Werkzeug diese in der Regel also doppelt. Das bedeutet höhere Fertigungszeiten und ggf. miserable Kantenbearbeitung. Leider entfernen nur wenige Software Tools diese überflüssigen Kanten überhaupt.
- Bevorzugte Richtung der Schachtelung einstellen (in Anbetracht einer möglichen Maserung der Grundplatte. Insbesondere bei Holzbauteilen relevant)
- einzelne Teile in ihrer Position arretieren. Nicht für jedes Bauteil ist es gut, wenn dieses in jedem beliebigen Winkel gedreht werden kann
- Möglichkeit, das Nesting für verschieden geformte Grundplatten (bzw. Restplatten) zu berechnen. Die meisten Nesting-Tools können nur mit Rechteckplatten arbeiten. Während der Herstellung von Dingen bleiben regelmäßig Verschnittplatten mit ungewöhnlichen Geometrien übrig. Durch ein effizientes Nesting-Programm, ggf. mit einer Verschnittplattenbibliothek, könnten diese ebenso prima genutzt werden.
- Texturen, Linien und Layer beachten. Häufig bestehen Plattenteile nicht nur aus Außenkonturen, welche geschnitten werden. Vielmehr sind häufig Markierungen, Prägungen, Linien, Schriftzüge, Embleme oder Texturen zu finden, die auf den Teileoberflächen entsprechend positioniert sind. In vielen Fällen gibt es durch Nesting-Software hierdurch hohen Informationsverlust, weil diese Elemente schlichtweg nicht beachtet werden.
- ...

Nesting Software

Folgende Tools haben wir im Laufe der Jahre als Liste zusammengetragen.

Nesting für beliebige Formen

Open Source

- [Cutting Optimizer \(Nesting\)](#) (InkScape Extension)
- [Deepnest](#) (works fine for rectangle nesting!)
- [Svgnest](#)
- [Arch Nest](#) (FreeCAD Extension)
- [KeepsakeAutomation/Functional-modules-for-Laser-Cutting-Nesting](#) (FreeCAD *.fcmacro Nesting generator)
- [markfink/nest2D](#)
- [github.com/lryan599/2DNesting](#)

Kommerziell

- [almaCAM](#)
- [angelblade CAM](#)
- [Astra R-Nesting](#)
- [AutoDesk TruNest](#)
- [AutoNEST](#)
- [blank slate](#)
- [BobCAD-CAM Nesting](#)
- [Cameo LaserDRAW](#)
- [Condacam LT](#)
- [CNCCut](#)
- [CutFab](#)
- [CutLeader](#)
- [eCut6](#) (Corel Draw Plugin)
- [Efficient Software NestFab](#)
- [FILOU Nest](#)
- [Gerber AccuNest](#)
- [Haitek FieryCut](#)
- [JETCAM](#)
- [NestMaker](#)
- [NestProfessor](#)
- [OptiNest](#)

- [Optitex Nesting](#)
- [powernestlib.com](#)
- [ProNest LTS/NestMaster](#)
- [sigmaNEST](#)
- [TANGENTA DG Nest Pro](#)
- [Vectric VcarvePro](#)
- [Vectric Aspire](#)
- [Vintech RCAM-Pro](#)

Guillotine Stock Problem (Nesting für Recktecke)

Suchworte: Cutting Optimizer, Panel Optimizer, strip nesting, rectangle nesting, strip packing, Guillotine cutting

Rechtecke zu verschachteln ist deutlich einfacher als das Verschachteln beliebiger Formen. Natürlich können alle oben gelisteten Nestingprogramme auch simple Rechtecke schachteln. Folgende weitere, speziell für Rechteckzuschnitte entworfene Nesting Tools gibt es außerdem:

Open Source

- github.com/jasonrhansen/cut-optimizer-2d-ng (based on cut-optimizer-2d-web, which is based on cut-optimizer-2d (library))
- [cottonrays/cuttlefish](https://github.com/cottonrays/cuttlefish) (less options than expected)
- pypi.org/project/opcut (has some strange bugs)
- [Oscar-Oliveira/SVG-Pattern-Editor](https://github.com/Oscar-Oliveira/SVG-Pattern-Editor) (not so well in handling)
- github.com/geri1701/freecut (no possibility to set back if sth. was entered wrong)
- github.com/lairdubois/lairdubois-opencutlist-sketchup-extension
- github.com/sbnedkov/cutlist (<https://www.razkroi.com>)
- github.com/natancl1/cutting_plan_linux (needs to be upgraded Python3)
- github.com/mru00/cutlet (hard to install because of JavaFX)
- github.com/jmcgill/cutlayout
- github.com/Hendrik-Jacobs/Cutting_Optimizer_v1.1 (Stern) (Windows only)

- Deepnest (see top)

Kommerziell

- cutmaster2d.com
- cutting-optimization.optimalprograms.com
- fastcut-optimizer
- rasterweq.com
- maxcutsoftware.com (has some free edition)
- tmachines.com/cutlogic-2d
- cutlistoptimizer.com (5 Berechnungen pro Tag)

Kostenfreie Closed Source

Nicht ganz dasselbe, aber ähnlich → [Bouwkamp Code](#)

Cutting Stock Problem (1D Nesting für Stabmaterial)

- <https://github.com/filipwodnicki/custo>
- <https://github.com/sr01/wood-cut-planner> (scheint nicht wirklich nutzbringend)

Zweckentfremdung: 3D-Nesting Software für 2D-Probleme nutzen

Die meisten Verschachtelungsprogramme in der Übersicht in [3D Bauteilverschachtelung / Plating / Nesting / Bin Packing](#) schachteln nicht im Vollvolumen, sondern auf der flachen Ebene. Falls man über geeignete 2D-Daten verfügt, welche sich beispielsweise durch Extrusion einfach in die Z-Dimension zu STL Dateien verwandeln lassen, können auch Nesting Tools wie Plater und Pack3D (und weitere) verwendet werden, um eigentliche zweidimensionale Probleme zu lösen. Im Anschluss an das Nesting müssen die geschachtelten 3D-Daten allerdings ebenso wieder auf einen 2D-Datensatz reduziert werden (Entfernen der dritten Dimension). Eine mögliche Prozesskette könnte zum Beispiel so aussehen:

Als Workflow dargestellt: SVG Teile in InkScape zeichnen → Teile in STL-Dateien konvertieren (z.B. [Paths to OpenSCAD](#)) → In einem 3D Nesting Programm schachteln (z.B. Plater) → Re-Import der geschachtelten STL-Datei als 2D-Datensatz via [Slic3r STL Input](#).

Version #1

Erstellt: 2026-04-16 15:47:47 CEST von Mario Voigt

Zuletzt aktualisiert: 2026-04-16 15:49:32 CEST von Mario Voigt