Architectural Blueprint for Phase 3: The Sequencer

Advanced A* Optimization for Non-Linear Masked Time Domains in Salvagnini P4 Kinematics

1. Executive Summary and Strategic Context

This research report articulates the mathematical and architectural foundation for "The Sequencer," the third phase of the proprietary control software development for the Salvagnini P4 Panel Bender. As the manufacturing sector migrates from high-volume, low-mix production to high-mix, "Batch One" and "Kit Production" models, the operational efficiency of automated cells is no longer defined solely by mechanical speed but by algorithmic intelligence.[1] The P4, a Tier 1 flexible manufacturing system, represents a significant capital investment—often retaining 40–50% of its value after five years—and its Return on Investment (ROI) is strictly correlated to the reduction of cycle time per part.[1]

The core technical challenge addressed herein is the optimization of the bending sequence in a non-linear cost domain. Unlike traditional serial manipulators where operation times are additive, the P4 utilizes a parallel actuation architecture. The Automatic Blankholder Adjustment (ABA) alters tool lengths simultaneously with the Manipulator's spatial reorientation of the sheet. This concurrency creates "Masked Time"—intervals where the cost of a slower operation totally eclipses the cost of a faster one.[1] Consequently, the Total Cycle Time (TCT) is governed by a $\max()$ function rather than a summation, rendering standard Traveling Salesman Problem (TSP) heuristics inadmissible or inefficient.

This report derives the rigorous Cost Function $f(n)$ for the A* Search Algorithm, specifically tailored to this non-linear environment. We establish the "Reverse Unfolding" state-space methodology as the optimal framework for collision avoidance and feasibility analysis.[3]

Furthermore, we derive a composite, admissible heuristic $h(n)$ comprising three distinct estimators: **Rotational Entropy** (kinematic lower bound based on spherical geometry), **Tooling Variance** (mechanical lower bound based on ABA dynamics), and **Grasp Fragmentation** (topological lower bound using the Largest Empty Rectangle algorithm to predict expensive "Repo" operations).

The architectural decisions detailed below effectively bridge the gap between theoretical combinatorial optimization and the physical realities of the P4-2516 and P4-2116 platforms [5], ensuring that the software delivers the mathematically guaranteed minimum cycle time.

2. The P4 Kinematic Model: Physics of the Optimization Domain

To develop a high-fidelity Sequencer, we must first mathematically formalize the machine's behavior. The Salvagnini P4 is not a generic robotic arm; it is a constrained Cartesian system with specific concurrent axes.

2.1 The Principle of Masked Time

In the context of the P4, "masked time" is the operational overlap between the manipulator's positioning actions and the blankholder's tooling adjustments. The P4-2525 and P4-xx16 models feature the ABA (Automatic Blankholder Adjustment), which adjusts the length of the hold-down tool in steps (typically 5mm increments) to accommodate the length of the next bend.[5]

Simultaneously, the manipulator—an electric or hydraulic carriage, depending on the specific P4 generation—must rotate the sheet metal (Coordinate θ) and translate it (Coordinates X, Y) to align the target bend line with the bending blades.[2]

Let σ_i represent the machine state at bend i, and σ_j the state at bend j. The transition δ_{ij} involves two primary time components:

1. **Kinematic Time (T_{kin}):** The time required to rotate and translate the sheet. This is a function of the manipulator's angular velocity ω and linear velocity v.

$$T_{kin}(\sigma_i, \sigma_j) = \max\left(\frac{|\theta_j - \theta_i|}{\omega_{manip}}, \frac{||P_j - P_i||}{v_{trans}}\right) + \tau_{settle}$$

2. **Tooling Time (T_{tool}):** The time required for the ABA to resize from length L_i to L_j.

$$T_{tool}(\sigma_i, \sigma_j) = \frac{|L_j - L_i|}{v_{ABA}} + \tau_{setup}$$

The fundamental cost equation for a transition in the P4 architecture is non-linear:

$$C_{trans}(\sigma_i, \sigma_j) = \max(T_{kin}(\sigma_i, \sigma_j), T_{tool}(\sigma_i, \sigma_j))$$

If $T_{kin} > T_{tool}$, the tool change is "masked" by the rotation. If $T_{tool} > T_{kin}$, the rotation is masked by the tool change. The Sequencer's objective is to find a path where the maximal envelope of these two functions is minimized.[1]

2.2 System Latency and Mechanical Constraints

The definition of these time components relies on precise hardware specifications.

- **Manipulator Dynamics:** The rotation unit typically operates with high acceleration. However, the rotational inertia of the sheet I_{sheet} varies with the part geometry. While often modeled as constant, advanced implementations may adjust ω_{manip} dynamically based on the part's current moment of inertia to prevent sheet slippage or deformation.[2] For the heuristic derivation, we assume a conservative constant ω_{max} (e.g., 90 deg/s for heavy sheets, higher for lighter ones).[8]
- **ABA Dynamics:** The ABA adjusts in discrete steps. The "zero set-up time" claim in marketing literature refers to the *inter-part* setup capability, but *in-cycle* adjustments consume real milliseconds.[5] The adjustment speed v_{ABA} is a critical constant. If the sequence requires expanding the tool from 500mm to 2000mm, and v_{ABA} is finite, this transition dominates the cost.
- **Bending Mechanics:** The bending action itself (the stroke of the upper and lower blades) takes a fixed duration τ_{bend} per hit, regardless of sequence. However, the approach and retract times can vary based on the "positive" (up) or "negative" (down) nature of the bend.[2]

2.3 The "Repo" Discontinuity

A third, strictly additive cost factor is the Repositioning ("Repo") operation. The manipulator grips the sheet at a specific location Γ. As the sheet is folded, flanges are created that may collide with the gripper or the machine frame. If the current grasp Γ becomes invalid for the next bend b_{next}, the machine must perform a Repo cycle:

1. Deposit sheet on table.
2. Release gripper.
3. Travel to new grasp location Γ'.
4. Engage gripper.
5. Lift sheet.

This operation effectively halts all masked parallelism. It introduces a massive penalty C_{repo}

(often 15–30 seconds).[9] A single unnecessary Repo can negate the efficiency gains of optimal routing. Therefore, the Cost Function must be:

f(n) = \sum_{k=1}^{N} \left

where $\mathbb{I}_{repo}^{k}$ is an indicator function that is 1 if a Repo occurs at step k, and 0 otherwise.

---

3. State Space Formalism: The Reverse Unfolding Paradigm

The choice of state space representation is the single most significant architectural decision in Phase 3. While the physical process is forward (flat $\rightarrow$ bent), the algorithmic process must be reverse (bent $\rightarrow$ flat).

3.1 Justification for Reverse Planning

Forward planning is computationally prohibitive due to the complexity of collision detection. To determine if a bend is feasible in forward mode, one must simulate the elastoplastic deformation of the metal and check if the resulting 3D flange intersects with the tool or the part itself.[4] This requires continuous collision detection (CCD) which is computationally expensive.

In contrast, **Reverse Unfolding** treats the finished part as the starting state. The operation "unfold bend b" simply restores the flange to the flat plane. If the flange, in its bent state, does not intersect any other geometry, it is a candidate for unfolding. This reduces the collision check to a static intersection test, which is orders of magnitude faster.[3] Furthermore, reverse unfolding naturally handles geometric precedence; if Flange A physically blocks Flange B, Flange B cannot be unfolded until Flange A is removed. This implicitly builds the dependency graph.

3.2 Feature Recognition and the Attributed Adjacency Graph (AAG)

To support reverse unfolding, the CAD model (STEP/IGES) is parsed into an **Attributed Adjacency Graph (AAG)**.

- **Nodes:** Represent planar faces of the sheet metal.
- **Edges:** Represent bend lines.
- **Attributes:** Each edge carries metadata: bend angle, radius, length, and normal vector direction (mountain/valley).[4]

From the AAG, we derive the **Precedence Graph**. A bend b_i precedes b_j if unfolding b_j causes a collision with b_i. This graph prunes the A* search space significantly. If b_i must be unfolded before b_j, any state where b_j is unfolded while b_i remains is unreachable and need not be generated.

3.3 Node Definition in A*

A state n in our search tree is defined by the tuple:

$$n = \langle \mathcal{G}_{rem}, \theta_{curr}, L_{ABA}, \Gamma_{pos} \rangle$$

1. $\mathcal{G}_{rem}$ **(Geometry):** A bitmask representing the set of bends *remaining* to be unfolded.
   - Start Node: $\mathcal{G}_{rem} = \{1, 1, \dots, 1\}$ (All bends present).
   - Goal Node: $\mathcal{G}_{rem} = \{0, 0, \dots, 0\}$ (Flat sheet).
2. θ_{curr} **(Orientation):** The current orientation of the part relative to the press centerline ($0^\circ, 90^\circ, 180^\circ, 270^\circ$).
3. L_{ABA} **(Tool State):** The current length configuration of the ABA. This allows precise calculation of ΔL for the next step.
4. Γ_{pos} **(Grasp):** The discrete index of the current gripper position on the sheet boundaries.

This state definition allows the A* algorithm to track not just the logical sequence of bends, but the physical state of the machine, enabling the accurate calculation of $g(n)$ (past cost) and $h(n)$ (heuristic future cost).

---

4. Derivation of the Admissible Heuristic $h(n)$

The A* algorithm is only as good as its heuristic. For the P4 Sequencer, $h(n)$ must estimate the remaining time to unfold all bends in $\mathcal{G}_{rem}$. To ensure optimality, $h(n)$ must be **admissible** (it must strictly underestimate or equal the true cost $h^*(n)$). If $h(n) > h^*(n)$, A* may converge on a suboptimal solution.

Given the cost function $C = \max(T_{kin}, T_{tool}) + C_{repo}$, deriving a tight lower bound is complex. We decompose $h(n)$ into three independent lower-bound functions:

1. $h_{rot}(n)$**:** The minimum kinematic time required.
2. $h_{tool}(n)$**:** The minimum tooling adjustment time required.
3. $h_{grasp}(n)$**:** The minimum Repo penalty.

The composite heuristic is:

$$h(n) = \max(h_{rot}(n), h_{tool}(n)) + h_{grasp}(n) + \sum_{b \in \mathcal{G}_{rem}} \tau_{bend}(b)$$

We will now rigorously derive each component.

---

5. Component 1: Rotational Entropy (h_{rot})

This heuristic captures the "kinematic disorder" of the remaining bends. If all remaining bends are on the same side of the sheet, the rotational cost is zero. If they are distributed across all four sides, the machine must perform rotations.

5.1 The Gaussian Sphere Abstraction

We utilize the concept of the **Gaussian Sphere** from differential geometry to map the normal vectors of the bend lines.[12] In the planar world of sheet metal, this simplifies to the **Gaussian Circle**. Each bend $b \in \mathcal{G}_{rem}$ has a normal vector n_b. For a standard rectangular panel:

- Top Edge $\rightarrow n = (0, 1)$ (0°)
- Right Edge $\rightarrow n = (1, 0)$ (90°)
- Bottom Edge $\rightarrow n = (0, -1)$ (180°)
- Left Edge $\rightarrow n = (-1, 0)$ (270°)

The manipulator must align each of these normals with the machine's press vector $P = (0, 1)$ sequentially.

5.2 Spherical TSP and Minimum Spanning Trees

The problem of minimizing rotation time is equivalent to the **Spherical Traveling Salesman Problem (TSP)**: find the shortest path on the circle that visits the normals of all remaining bends.[14] Solving TSP is NP-Hard. For a heuristic, we need a computationally cheap **lower bound**.

Derivation:

Let Φ_{rem} be the set of unique angular orientations required by the bends in $\mathcal{G}_{rem}$, plus the current orientation θ_{curr}.

The absolute minimum rotation required to visit a set of points on a circle is defined by the **Shortest Angular Cover**. This is the length of the smallest arc that contains all points in Φ_{rem}.

However, since the P4 can rotate clockwise or counter-clockwise, and the sequence matters, a tighter bound is the **Minimum Spanning Tree (MST)** on the angular graph.

For the discrete case of $\{0, 90, 180, 270\}$, the MST calculation is trivial.

If the set contains $\{0, 180\}$, the minimum rotation is 180° (or $90 + 90$). If it contains $\{0, 90, 270\}$, the minimum is 180° (e.g., $90 \to 0 \to 270$ spans 180°).

Mathematical Definition of h_{rot}:

Let Θ_{req} be the set of required angles.

$$h_{rot}(n) = \frac{\text{Range}(\Theta_{req} \cup \{\theta_{curr}\})}{\omega_{manip}}$$

where $\text{Range}()$ calculates the minimal angular sector covering the set. This is strictly admissible because the manipulator *must* physically traverse these angles to present the bends to the tool. It assumes optimal ordering (no backtracking), which is the definition of a lower bound.

Advanced Insight: In "Batch One" production, we might encounter arbitrary angles (e.g., diagonal bends). In this case, the Gaussian Circle mapping remains valid. The heuristic simply computes the bounding arc of the required normals. If diagonal bends exist (e.g., 45°), the entropy increases, and the heuristic accurately reflects the increased kinematic cost.[13]

6. Component 2: Tooling Variance (h_{tool})

This heuristic estimates the cost driven by the ABA. In highly variable "Kit Production" [16], the ABA often becomes the bottleneck, as the tool length must change radically between parts or even between bends on the same part.

6.1 The 1D Tooling Space

The ABA is a linear axis. Its state is defined by a scalar Length $L \in [L_{min}, L_{max}]$.

The remaining bends $\mathcal{B}_{rem}$ require a set of discrete tool lengths $\Lambda = \{l_b \mid b \in \mathcal{G}_{rem}\}$.

Current length is L_{curr}.

We must visit every length in Λ. This corresponds to a **1D TSP**.

For points on a line, the TSP is solved exactly by traversing from the start point L_{curr} to the extremum in one direction (min or max) and then to the other extremum.

However, for *admissibility* under the max() operator of masked time, we must be careful. We are looking for the *minimum mandatory* tool movement that cannot be avoided regardless of the rotation sequence.

6.2 The Range Heuristic

The absolute minimum distance the tool *must* travel is the difference between the maximum and minimum required lengths in the remaining set, expanded to include the current length.

Let $\Lambda^+ = \Lambda \cup \{L_{curr}\}$.

$$\Delta L_{min} = \max(\Lambda^+) - \min(\Lambda^+)$$

This quantity represents the "span" of the tooling requirements. It assumes the optimal traversal (monotonic increase or decrease).

$$h_{tool}(n) = \frac{\Delta L_{min}}{v_{ABA}}$$

Why this is admissible:

Even if the optimal sequence requires "zig-zagging" (e.g., $500 \to 1000 \to 500$), the *range* ($1000 - 500 = 500$) is a lower bound on the total travel ($500 + 500 = 1000$). Thus, $h_{tool} \le h^*_{tool}$.

By combining this with h_{rot} using $\max()$, we capture the "bottleneck" nature of the masked time. If the tool variance is high but rotational entropy is low, h_{tool} will drive the heuristic value, correctly identifying that the ABA is the limiting factor.

Impact of ABA Segmentation:

The ABA adjusts in 5mm steps. The heuristic should ideally account for this quantization.

$$h_{tool}(n) = \frac{\lceil (\max(\Lambda^+) - \min(\Lambda^+))/\text{step} \rceil \times \text{step}}{v_{ABA}} + \epsilon_{latency}$$

This refinement ensures the heuristic respects the mechanical granularity of the Salvagnini P4.[5]

---

7. Component 3: Grasp Fragmentation and the LER Algorithm

This is the most critical and mathematically complex component. A "Repo" operation is a discontinuous cost penalty ($C_{repo} \approx 20s$) that breaks the flow of production. Avoiding Repos is the primary strategic imperative of the Sequencer.

7.1 The Topological Grasp Problem

The manipulator requires a simplified "Robot Grasp Location" to handle the sheet. This location must remain valid for a sequence of bends. A location is valid for bend b if the gripper, when placed there, does not collide with the tools or the flanges generated by b.[9] Let $V(b) \subset \mathbb{R}^2$ be the set of valid gripper coordinates (on the sheet surface) for bend b. For a sequence $S = \{b_1, \dots, b_k\}$ to be feasible without a Repo, there must exist a common intersection:

$$I_S = \bigcap_{i=1}^{k} V(b_i) \neq \emptyset$$

If $I_S = \emptyset$, the sequence S is **Grasp Fragmented**, and at least one Repo is mandatory.

7.2 The Largest Empty Rectangle (LER) Implementation

Computing the intersection of arbitrary polygons $V(b)$ is computationally expensive ($O(N^2 \log N)$). To perform this inside the heuristic function (which runs thousands of times per second), we need a faster approximation.

We model the sheet as a rectangular domain D. Each bend b introduces "Forbidden Zones" Z_b (where the gripper cannot go).

$$V(b) = D \setminus Z_b$$

The intersection condition becomes:

$$I_S = D \setminus \bigcup_{b \in \mathcal{G}_{rem}} Z_b$$

We need to find if there exists a rectangle R_{grip} of the gripper's dimensions (w_g, h_g) that fits inside I_S. This is the **Largest Empty Rectangle (LER)** problem.[17]

Algorithm Adaptation for Heuristic h_{grasp}:

1. **Projection:** For all remaining bends in $\mathcal{G}_{rem}$, project their Forbidden Zones onto the 2D plane of the sheet. These zones are typically rectangular strips near the bend lines.
2. **Aggregation:** Treat these zones as obstacles.
3. **LER Solve:** Use the maximal empty rectangle algorithm (based on determining the largest rectangle of 0s in a binary matrix, or using a sweep-line algorithm with segment trees) to find the largest available area R_{max} .
4. **Feasibility Test:**
   - If $Area(R_{max}) \geq Area(R_{grip})$, then a single grasp *might* exist. $h_{grasp} = 0$.
   - If $Area(R_{max}) < Area(R_{grip})$, then it is mathematically impossible to hold the sheet for all remaining bends. A Repo is guaranteed.
   - $h_{grasp}(n) = C_{repo}$.

Complexity:

The sweep-line LER algorithm runs in $O(n \log n)$, where n is the number of bends. This is fast enough for heuristic evaluation.

Admissibility: This heuristic is strictly admissible because it only applies the penalty when a solution without a Repo is provably impossible. It does not predict *potential* Repos (which might be avoided), only *certain* ones.

7.3 Pairwise Disjoint Optimization

For extreme speed, we can use a weaker but $O(N^2)$ pre-check.

Iterate through all pairs $(b_i, b_j) \in \mathcal{G}_{rem}$.

If $V(b_i) \cap V(b_j) = \emptyset$ (i.e., the valid zones for two bends are disjoint), then they cannot share a grasp.

If such a pair exists in $\mathcal{G}_{rem}$, $h_{grasp} = C_{repo}$.

This "Pairwise Disjoint" check is instant if intersections are pre-computed, providing a rapid lower bound before running the full LER.

---

8. Integrated Cost Function and Search Logic

We combine the derived components into the final architectural definition.

8.1 The Unified Heuristic

$$h(n) = \underbrace{\max\left(\frac{\text{Span}(\Theta_{rem})}{\omega_{manip}}, \frac{\text{Range}(\Lambda_{rem})}{v_{ABA}}\right)}_{\text{Masked Time Lower Bound}} + \underbrace{K_{repo} \times C_{repo}}_{\text{Grasp Penalty}} + \underbrace{\sum \tau_{bend}}_{\text{Process Base}}$$

where $K_{repo} = 1$ if $\text{LER}(\mathcal{G}_{rem}) < \text{ Threshold}$, else 0.

8.2 A* Search Configuration

- **Algorithm:** We employ **IDA*** (Iterative Deepening A*) or **SMA*** (Simplified Memory Bounded A*) depending on the memory constraints of the specific P4 controller (some legacy controllers have limited RAM). IDA* is generally preferred for this domain as the tree depth is fixed (number of bends).[19]
- **Tie-Breaking:** The "Plateau Problem" is common in P4 optimization, where many

sequences have identical theoretical costs (e.g., when ABA time completely masks rotation time for several steps).

- *Rule:* If $f(n_1) = f(n_2)$, prefer the node with higher **Grasp Stability** (larger LER area). This biases the search towards "robust" solutions that are less likely to fail due to sensor tolerances in the physical machine.

8.3 State Pruning via Domination

To handle the combinatorial explosion (Factorial complexity $N!$), we implement **Domination Pruning**.

Two states n_A and n_B are comparable if:

1. $\mathcal{G}_{rem}^A = \mathcal{G}_{rem}^B$ (Same remaining geometry).
2. $\theta_{curr}^A = \theta_{curr}^B$ (Same orientation).
3. $L_{ABA}^A = L_{ABA}^B$ (Same tool length).

If $g(n_A) < g(n_B)$, does n_A dominate n_B? Not necessarily, because n_B might have a better grasp position Γ_B that avoids a future Repo.

- *Strict Domination:* n_A dominates n_B if $g(n_A) \leq g(n_B)$ AND $V(\Gamma_A) \supseteq V(\Gamma_B)$.
  - (i.e., Cost is lower, and the grasp is strictly more versatile or equal).
  - If dominated, prune n_B.

---

9. Hardware-Software Integration and ROI Implications

This mathematical framework translates directly into measurable industrial performance.

9.1 Cycle Time and ROI

The P4's market value is driven by its productivity. By optimizing the masked time effectively, we reduce the "dead time" between bends.

- **Linear vs. Masked:** A linear sequencer would sum $T_{rot} + T_{tool}$. Our masked sequencer sums $\max(T_{rot}, T_{tool})$.
- **Impact:** For a typical panel with 10 bends, where $T_{rot} \approx 1s$ and $T_{tool} \approx 1s$, a linear

approach yields 20s transition time. Masked approach yields 10s. This 50% reduction in transition time can lead to a 15-20% reduction in TCT (Total Cycle Time), directly increasing the parts-per-hour output and improving the customer's ROI timeframe significantly.[1]

9.2 LINKS and Industry 4.0

The calculated $f(n)$ and $h(n)$ values are not discarded. They are fed into the **LINKS** IoT monitor (Salvagnini's Industry 4.0 solution).[21]

- **Predictive Maintenance:** If the *actual* cycle time consistently exceeds the *predicted* $f(Start)$, it indicates mechanical degradation (e.g., ABA friction increasing, manipulator belt slack).
- **Feedback Loop:** The "settling time" parameters τ_{settle} in our cost function can be dynamically tuned based on real-time feedback from the machine, creating a self-optimizing control loop.

---

10. Conclusion

The "Sequencer" for the Salvagnini P4 is not merely a sorting algorithm; it is a multi-objective optimization engine operating in a non-linear cost landscape. This report has established that:

1. **Masked Time** requires a heuristic based on the $\max()$ operator, utilizing Gaussian Spheres for rotational bounds and 1D TSP ranges for tooling bounds.
2. **State Management** must utilize Reverse Unfolding to effectively handle the geometric constraints of sheet metal forming.
3. **Grasp Planning** is the highest-risk element, solvable via the admissible Largest Empty Rectangle heuristic to predict and penalize Repo operations.

By implementing this architecture, we ensure that the P4 operates at its theoretical kinematic limit, justifying its status as a Tier 1 Flexible Manufacturing System.

Key Data Tables

Table 1: Admissible Heuristic Components

Component	Physical Proxy	Mathematical Model	Complexity	Admissibility Logic

Rotational Entropy (h_{rot})	Manipulator Rotation	Spherical TSP / Arc Span	$O(N)$	Minimal angular sector required to cover normals.
Tooling Variance (h_{tool})	ABA Adjustment	1D Range / 1D TSP	$O(N)$	Minimal linear travel to cover tool length extrema.
Grasp Frag. (h_{grasp})	Repo Operation	Largest Empty Rectangle	$O(N \log N)$	Adds penalty only if intersection of valid zones is null.

Table 2: P4 Kinematic Parameters for Cost Function

Parameter	**Symbol**	**Typical Value**	**Source**	**Impact on f(n)**
Manipulator Speed	ω_{manip}	90–135 deg/s	[5]	Denominator for h_{rot}.
ABA Speed	v_{ABA}	~200 mm/s	[7]	Denominator for h_{tool}.
Repo Penalty	C_{repo}	20–30 sec	[10]	Step discontinuity in $g(n)$.
Bend Time	τ_{bend}	~2 sec	[23]	Additive constant.

Citations:.[1]

Works cited

1. Panel Bending Tools - JEELIX, accessed February 3, 2026, https://www.jeelix.com/panel-bending-tools/
2. Panel Bender -P4 - MachineryHost, accessed February 3, 2026, https://f.machineryhost.com/71b2fab0316bfa7fa24ce0a0e5460bad/Brochure%20-%20Panel%20Bender%20-P4.pdf
3. On sheet metal unfolding. Part 6: bend sequences | by Analysis Situs - Medium, accessed February 3, 2026, https://medium.com/@analysis-situs/on-sheet-metal-unfolding-part-6-bend-sequences-11d06840c5e8
4. Our paper in Computer-Aided Design and Applications | by Analysis Situs - Medium, accessed February 3, 2026, https://analysis-situs.medium.com/our-paper-in-computer-aided-design-and-applications-19b7b6292e6a
5. The automatic hybrid adaptive Panel Bender. - Sanson Machinery Group, accessed February 3, 2026, http://www.sansonmachinery.com/wp-content/uploads/2013/12/P4Xe-UK-3962020312.pdf
6. Panel bender cuts production costs for Hydram - Machinery Market News, accessed February 3, 2026, https://www.machinery-market.co.uk/news/8375/Panel-bender-cuts-production-costs-for-Hydram
7. The flexible bending solution. - Sp-Tech, s.r.o., accessed February 3, 2026, https://www.sp-tech.cz/static/soubory/stranka-71/prospekt-salvagnini-p2-152.pdf
8. p4-benders.pdf - Empire Machinery, accessed February 3, 2026, https://www.empire-machinery.com/wp-content/uploads/2021/06/p4-benders.pdf
9. Automated Process Planning for Sheet Metal Bending Operations - Carnegie Mellon University's Robotics Institute, accessed February 3, 2026, https://www.ri.cmu.edu/pub_files/pub1/kim_kyoung_k_1998_1/kim_kyoung_k_1998_1.pdf
10. MAY 2021 Surplus Record Machinery & Equipment Directory | PDF | Electric Motor - Scribd, accessed February 3, 2026, https://www.scribd.com/document/503588262/MAY-2021-Surplus-Record-Machinery-Equipment-Directory
11. Y.-Q. Liu's research works | Huazhong University of Science and Technology and other places - ResearchGate, accessed February 3, 2026, https://www.researchgate.net/scientific-contributions/Y-Q-Liu-2031227489
12. Complex 3D Shape Recovery using Hybrid Geometric Shape, accessed February 3, 2026, https://kops.uni-konstanz.de/server/api/core/bitstreams/9aa084fb-f3f2-4630-bb22-20648040874f/content
13. Improvement of piecewise developable surface approximation algorithm based on Gauss image - SPIE Digital Library, accessed February 3, 2026,

https://www.spiedigitallibrary.org/conference-proceedings-of-spie/13664/1366413/Improvement-of-piecewise-developable-surface-approximation-algorithm-based-on-Gauss/10.1117/12.3070652.full
14. The application of ant colony optimization in the solution of 3D traveling salesman problem on a sphere, accessed February 3, 2026, https://earsiv.kmu.edu.tr/server/api/core/bitstreams/518ece52-0167-4075-a36a-a62ca2288226/content
15. GENETIC ALGORITHM BASED SOLUTION FOR TSP ON A SPHERE Aybars Uğur, Serdar Korukoğlu, Ali Çalışkan, Muhammed Cinsdikici, Ali - Semantic Scholar, accessed February 3, 2026, https://pdfs.semanticscholar.org/f9ab/10a7754c65b1a4e0867494e296bb942eb103.pdf
16. P4 Abt: Range of Automatic Panel Benders. More Flexible, More Versatile, More Productive | PDF | Sheet Metal - Scribd, accessed February 3, 2026, https://www.scribd.com/document/342321104/P4-ENG-3962020307-1006
17. A branch and bound algorithm for the strip packing problem | Request PDF - ResearchGate, accessed February 3, 2026, https://www.researchgate.net/publication/226128373_A_branch_and_bound_algorithm_for_the_strip_packing_problem
18. Computing the Largest Empty Rectangle - SIAM.org, accessed February 3, 2026, https://epubs.siam.org/doi/10.1137/0215022
19. Admissible Heuristics for Optimal Planning - IDA.LiU.SE, accessed February 3, 2026, https://www.ida.liu.se/divisions/aiics/publications/AIPS-2000-Admissible-Heuristics-Optimal.pdf
20. A Parallel CPU-GPU Framework for Batching Heuristic Operations in Depth-First Heuristic Search - arXiv, accessed February 3, 2026, https://arxiv.org/html/2507.11916v2
21. HIGHLIGHTS, accessed February 3, 2026, https://irp-cdn.multiscreensite.com/d90c5f8c/files/uploaded/Salvagnini%20Product%20overview%20Highlights.pdf
22. Intelligent system for generating and executing a sheet metal bending plan - Google Patents, accessed February 3, 2026, https://patents.google.com/patent/US20040019402A1/en
23. Lean & Compact Panel Bender - Salvagnini P2, accessed February 3, 2026, https://www.salvagninigroup.com/en-US/products/panel-benders/P2
24. Panel Benders - Empire Machinery, accessed February 3, 2026, https://www.empire-machinery.com/wp-content/uploads/2021/06/p1-benders.pdf
25. Collision-free tool orientation optimization in five-axis machining of bladed disk, accessed February 3, 2026, https://academic.oup.com/jcde/article-pdf/2/4/197/33214926/j.jcde.2015.06.001.pdf
26. What is Sheet Metal Bending - A comprehensive guide - Jiga, accessed February 3, 2026, https://jiga.io/articles/sheet-metal-bending/

27. Sheet Metal Bending: Basics, Types and Design Considerations - Chiggo, accessed February 3, 2026, https://chiggofactory.com/basics-types-and-design-considerations-of-sheet-metal-bending/
28. Fluid Power with Applications Anthony Esposito Seventh Edition - ElCoM, accessed February 3, 2026, https://elcom-team.com/Subjects/%D8%A7%D9%86%D8%B8%D9%85%D8%A9%20%D9%87%D9%8A%D8%AF%D8%B1%D9%88%D9%84%D9%8A%D9%83%D9%8A%D8%A9/hydro.pdf