Existence and Construction of Reel Strips from Mathematical Specifications

Christian Fenner
Independent Researcher
Las Vegas, NV
fenner.d.christian@gmail.com

August 28, 2026

Contents

Abstract

The reel strip design problem asks whether a slot machine's symbol sequences can be constructed from mathematical specifications rather than found by search. We prove that they can. A rearrangement invariant decomposes the per-symbol design space into three sequentially decoupled layers (hit rate, RTP contribution, and payout volatility), each with a characterized attainable set. A co-location matrix captures the order-2 cross-symbol window statistics that control total hit rate, with a reachable by mixing achievable set established by exact splice linearity. The achievable set of $(H, \mathrm{CV}_{\mathrm{win}})$ pairs at fixed counts is reachable by mixing: any interior point is reached by mixing extreme-point strips, with precision linear in strip length. Payout volatility decomposes exactly into a hit-rate component and a conditional win-shape component $\mathrm{CV}_{\mathrm{win}}$, targetable through clustering distribution, directed filler arrangement, and label permutation at frozen RTP and hit rate. The main result is a full construction theorem: for any per-symbol targets and any global targets (RTP, hit rate, volatility) in a characterized achievable set, with constructibility decidable from the co-location matrix before any strip is built, a reel strip realizing the full specification exists and can be constructed deterministically. Wild symbols require only a linear correction. Scatter symbols have count probabilities characterized by the Poisson-binomial distribution with multilinear achievable ranges. Symbols carrying numerical values (cash scatters, multiplier wilds) are handled by a linear system that separately targets per-count expected values with full distributional control. The framework supports multi-state games as Markov chains, with each state's reelset constructed independently and aggregate metrics computed from the stationary distribution. A worked example constructs a 5×3 ways-pay base game with compound wild multipliers, scatter triggers, and mixed paying depths, targeting RTP = 64%, hit rate = 1/4, and $\mathrm{CV}_{\mathrm{win}} = 15$ simultaneously through three sequentially decoupled controls in under 17 seconds on commodity hardware.

1 Introduction

A slot machine is a set of independently spinning reels, each displaying a window of symbols. The cyclic sequence of symbols on each reel, the *reel strip*, determines every mathematical property of the game: how often the player wins, how much the house retains, and how the wins are distributed between frequent small payouts and rare large ones. These properties emerge from the combinatorial interaction of the strips across reels over an infinite horizon, and casinos depend on their exactness. The game mathematician's task is to design strips that hit precise regulatory and business targets while delivering an experience that makes losing money enjoyable, a tension between mathematical constraint and player psychology that the existing literature has addressed only partially.

The reel strip design problem asks: given a complete set of performance targets at the per-symbol, per-event level, does a reel strip exist that realizes all targets simultaneously, and can it be constructed?

The existing literature formulates this as a global optimization problem. The available treatments of slot mathematics [16] are descriptive rather than constructive. Balabanov et al. [2] apply Genetic Algorithms to search for a strip achieving a target total RTP, using Monte Carlo simulation as the fitness function. The same group [3] extends this with Discrete Differential Evolution optimizing RTP, prizes equalization, and symbol diversity as a multi-criteria objective. Keremedchiev et al. [4] replace Monte Carlo with exact full-cycle computation. Kamanas et al. [5], from a separate research group, introduce Variable Neighborhood Search with two local search operators, achieving the current state of the art. As of 2025, the same group continues to present VNS-based RTP optimization as the frontier of the field [6], with separate control of hit rate and volatility listed as open future work. Separately, Groote et al. [7] model slot machines as probabilistic process specifications and compute exact RTP via quantitative model checking — rigorous evaluation of an existing machine, not construction from specifications. In all optimization approaches, the target is a single number or a weighted multi-criteria objective, and the method is iterative search over symbol distributions. Kamanas et al. note that their algorithm's behavior "in the case where special symbols appear in reels (e.g., wild, scatter) is also unknown."

We present the first constructive theory of reel strip design. The framework decomposes the design space into functionally decoupled controls that target global RTP, total hit rate, and total volatility simultaneously. This solves the problem the existing literature addresses, while extending control to the per-symbol, per-event level: individual RTP contributions, individual hit-rate frequencies, near-miss probabilities, and per-event volatility profiles, all specified independently and all guaranteed constructible. Every feasibility bound is computable from the specification alone. When targets are feasible, a reel strip realizing the full specification exists and can be constructed deterministically. No simulation is required; targeted refinement replaces undirected search.

The key tool is a rearrangement invariant (Theorem 2.7) that decomposes the per-symbol design space into three sequentially decoupled layers, each with a characterized attainable set. At the global level, the total RTP is targeted via filler count allocation (with existence by the intermediate value theorem on a convex polytope), the total hit rate is targeted via the *co-location matrix* (an order-2 window statistic with provable bounds on the uncaptured higher-order residual), and the total volatility is targeted by selecting the operating point on the RTP iso-surface that achieves the desired variance, with the co-location matrix realizing the hit rate at that point. The following are the specific contributions.

Contributions.

1. A rearrangement invariant and two structural consequences that decouple hit rate from RTP (Section 2).

2. A three-layer independence theorem with full characterization of the attainable set per layer, at the per-symbol, per-event level (Section 3).

3. An existence and constructibility theorem for the reel strip skeleton from arbitrary feasible per-symbol specifications (Section 4).

4. Multilinearity of RTP across reels, with a convex-path existence proof for any target RTP within a pre-computable band (Section 5).

5. The co-location matrix as a complete order-2 characterization of cross-symbol window statistics, with reachability of the achievable set established by exact splice linearity (Section 6). The $E[c^2]$ lattice provides a closed-form residual for fine volatility control.

6. Hit-rate targeting via the co-location matrix at frozen RTP, with the order-3+ residual measured at $\sim 10^{-4}$ on the worked example and correctable by splice (Section 7).

7. Payout volatility decomposition into hit-rate and conditional win-shape ($\mathrm{CV}_{\mathrm{win}}$) components, with exact $\mathrm{CV}_{\mathrm{win}}$ computation from design parameters and three per-reel controls at frozen RTP and hit rate (Section 8).

8. A full construction theorem: for any per-symbol targets (p, c, τ^2) within their attainable ranges and any global targets (RTP, hit rate, volatility) in a characterized achievable set, with constructibility decidable from the co-location matrix before any strip is built, a reel strip realizing the full specification exists and can be constructed deterministically (Section 9).

9. A linear system for value-bearing symbols (symbols carrying numerical values such as cash amounts or multipliers), independently targeting per-count expected values with $n - K$ free parameters for distributional control (where K is the number of count levels) (Section 11).

10. A complete scatter theory with Poisson-binomial count probabilities, multilinear achievable ranges, rational-function constrained ranges, and additive RTP decomposition with the ways-pay system (Section 12).

11. A worked example constructing a complete 5×3 ways-pay game with compound wild multipliers, targeting $(\mathrm{RTP}, \text{hit rate}, \mathrm{CV}_{\mathrm{win}}) = (64\%, 1/4, 15)$ simultaneously through three sequentially decoupled controls, verified by exact enumeration on commodity hardware (Section 15).

2 Definitions and the Rearrangement Invariant

Let Ω denote a finite set of symbols and let R denote the number of reels in a slot game.

Definition 2.1 (Reel Strip)**.** A *reel strip* for reel i is a cyclic sequence $\mathbf{r}_i = (r_{i,1}, r_{i,2}, \ldots, r_{i,L_i})$ of symbols drawn from Ω, where L_i is the *reel length*. Indices are taken modulo L_i.

Definition 2.2 (Spin)**.** A *spin* on reel i selects a position $j \in \{1, \ldots, L_i\}$ uniformly at random. The reels are sampled independently. By convention, the selected position j corresponds to the top entry of the display window.

Definition 2.3 (Window)**.** The *window* at position j on reel i with display height W is the multiset of W consecutive symbols $\{r_{i,j},\ r_{i,j+1},\ \ldots,\ r_{i,j+W-1}\}$ (indices mod L_i). There are exactly L_i distinct windows on reel i.

For a symbol $s \in \Omega$ on reel i, let $n_i^{(s)}$ denote the number of positions occupied by s on the strip (the *stop count*).

Definition 2.4 (Window Coverage)**.** Let $a_i^{(s)}$ denote the number of windows on reel i that contain at least one instance of symbol s. The *window coverage* is:

$$p_i^{(s)} = \frac{a_i^{(s)}}{L_i}. \tag{1}$$

This is the probability that a uniformly sampled window contains symbol s.

Definition 2.5 (Conditional Count)**.** The *conditional count* of symbol s on reel i is the expected number of instances of s visible in a window, given that at least one is visible:

$$c_i^{(s)} = \frac{W \cdot n_i^{(s)}}{a_i^{(s)}}. \tag{2}$$

The numerator $W \cdot n_i^{(s)}$ is the total number of *sightings* of s across all L_i windows: each of the $n_i^{(s)}$ stops appears in exactly W windows (those whose top position falls within W positions above the stop), regardless of how the stops are arranged on the strip.

Definition 2.6 (Symbol Percent and Mean Count)**.** The *symbol percent* of s on reel i is the fraction of reel positions occupied by s:

$$\mathrm{sp}_i^{(s)} = \frac{n_i^{(s)}}{L_i}. \tag{3}$$

The *mean count* is the expected number of instances of s in a uniformly sampled window:

$$m_i^{(s)} = W \cdot \mathrm{sp}_i^{(s)} = \frac{W \cdot n_i^{(s)}}{L_i}. \tag{4}$$

Since sp depends only on the stop count n and the reel length L, and W is a fixed display parameter, m is determined entirely by the composition of the strip, not the arrangement of symbols within it.

Theorem 2.7 (Rearrangement Invariant)**.** *The mean count $m = p \cdot c$, and this product is invariant under any rearrangement of the stops of s on the strip. Rearrangement trades p against c along the rectangular hyperbola $p \cdot c = m$.*

Proof. From Definitions 2.4, 2.5, and 2.6:

$$p \cdot c = \frac{a}{L} \cdot \frac{Wn}{a} = \frac{Wn}{L} = m. \tag{5}$$

The quantity a (the number of non-empty windows) cancels. Since $m = Wn/L$ depends only on n, W, and L, none of which change when stops are repositioned, the product $p \cdot c$ is invariant. □

Rearranging the stops of a symbol changes which windows contain it (a, and hence p) and how many copies appear in each non-empty window (c). Clustering stops increases c (more per window) while decreasing p (fewer distinct windows reached). Spreading stops decreases c toward 1 while increasing p toward Wn/L. The product is always m.

2.1 Game Evaluation

A slot game evaluates the outcome of a spin by comparing the symbols visible in each reel's window against a *paytable*. We consider the standard *ways* evaluation, in which every combination of symbol positions across reels constitutes a potential winning alignment.

Definition 2.8 (Paytable and OAK Ladder)**.** A *paytable* assigns to each symbol $s \in \Omega$ a sequence of pay values $v_2^{(s)}, v_3^{(s)}, \ldots, v_R^{(s)}$, where $v_k^{(s)}$ is the payout for a k-of-a-kind (kOAK) win. Wins are *left-anchored*: a kOAK of symbol s requires s to be present in the window on each of reels 1 through k, and absent from the window on reel $k+1$ (the *blocker* reel). An ROAK has no blocker.

Definition 2.9 (Ways Count)**.** In a ways game, the number of winning alignments for a kOAK of symbol s on a given spin is the product of the per-reel counts of s across the k paying reels. If reel i shows x_i instances of s in its window, the ways count is $\prod_{i=1}^{k} x_i$. The payout is $v_k^{(s)} \cdot \prod_{i=1}^{k} x_i$.

Definition 2.10 (Return-to-Player)**.** The *return-to-player* (RTP) is the expected payout per unit wagered over the infinite horizon. The total RTP is the sum of contributions from all symbols and all OAK levels:

$$\mathrm{RTP} = \sum_{s \in \Omega} \sum_{k=2}^{R} \mathrm{RTP}_k^{(s)}, \tag{6}$$

where $\mathrm{RTP}_k^{(s)}$ is the expected per-spin payout from kOAK wins of symbol s, divided by the wager.

2.2 Two Consequences of the Invariant

Proposition 2.11 (Ways RTP is First-Order)**.** *In a ways game, the RTP contribution of a kOAK win for symbol s is:*

$$\mathrm{RTP}_k^{(s)} = v_k^{(s)} \cdot \prod_{i=1}^{k} m_i^{(s)} \cdot (1 - p_{k+1}^{(s)}), \quad k < R, \tag{7}$$

and $\mathrm{RTP}_R^{(s)} = v_R^{(s)} \cdot \prod_{i=1}^{R} m_i^{(s)}$. *RTP depends only on per-reel means m_i (on the paying reels) and presence p_{k+1} (on the blocker reel). It never depends on within-window co-occurrence, on the conditional count c directly, or on any higher moment of the count distribution.*

Proof. Let X_i denote the count of symbol s in the window on reel i. The payout of a kOAK is $v_k \cdot \prod_{i=1}^{k} X_i$ when s is present on reels 1 through k and absent on reel $k+1$. Since the reels are sampled independently (Definition 2.2):

$$\mathbb{E}\left[\prod_{i=1}^{k} X_i\right] = \prod_{i=1}^{k} \mathbb{E}[X_i] = \prod_{i=1}^{k} m_i. \tag{8}$$

The blocker reel $k+1$ contributes the binary test $P(X_{k+1} = 0) = 1 - p_{k+1}$: it asks whether symbol s is present, not how many instances appear. Since each m_i is rearrangement-invariant (Theorem 2.7), the RTP is independent of how symbols are arranged within any reel. □

Corollary 2.12 (Reels 1–2 Are RTP-Free)**.** *No standard paytable awards a 1OAK. Therefore p_1 and p_2 never appear as blocker terms in (7), and reels 1 and 2 enter the RTP formula only through m_1 and m_2, which are rearrangement-invariant. The arrangement of symbols on reels 1 and 2 may be chosen freely without affecting any RTP quantity.*

Corollary 2.13 (Blocker-Safe Clustering)**.** *A filler symbol f with minimum pay depth d_f uses reel k as a blocker for its $(k-1)$OAK event, for each $k \in \{d_f+1, \dots, R\}$. Clustering f on reel i changes $p_i^{(f)}$ (at fixed $m_i^{(f)} = p_i \cdot c_i$), which changes the blocker term $(1 - p_i^{(f)})$ if i is a blocker reel for f. Consequently, clustering f on reel i is RTP-preserving if and only if i is not a blocker reel for f:*

- *Depth-2 fillers ($d_f = 2$) block on reels $3, 4, \dots, R$. Clustering is RTP-safe on reels 1–2.*
- *Depth-3 fillers ($d_f = 3$) block on reels $4, 5, \dots, R$. Clustering is RTP-safe on reels 1–3.*

In general, a filler with minimum pay depth d_f may be clustered freely on reels 1 through d_f without affecting any RTP quantity. This extends Corollary 2.12 from a per-reel statement to a per-symbol, per-reel statement: the relevant condition is not whether the reel is a blocker, but whether it is a blocker for the symbol being clustered.

Remark 2.14 (Constraint Chain on Blocker Reels)*.* On reels 1 and 2, which never serve as blockers, the window coverage p does not appear in the RTP formula. Both the stop count and the arrangement are free from RTP's perspective.

On reels $k \geq 3$, the coverage p_k enters (7) as the blocker term $(1-p_k)$. The constraints cascade:

1. **Fix** p: preserves the blocker contribution to RTP and preserves hit rate.
2. **Fix** c **(at that** p**)**: since $m = p \cdot c$, fixing both p and c fixes m, which preserves the paying-reel contribution to RTP. RTP is now fully frozen.
3. **Vary the count distribution**: at fixed p and c, different physical arrangements produce different distributions over $\{1, \dots, W\}$ with the same mean c. This changes per-event payout volatility while preserving both hit rate and RTP.

This is the mechanism by which all three layers remain functionally decoupled even on blocker reels.

3 Three-Layer Decomposition

The rearrangement invariant establishes that RTP is fixed under rearrangement. This section characterizes the full per-symbol design space: hit rate (Layer 1, controlled by coverage), RTP contribution (Layer 2, controlled by stop count at fixed coverage), and per-event volatility (Layer 3, controlled by count distribution at fixed coverage and stop count). These three layers are functionally decoupled at the per-symbol, per-event, per-reel level. Global game targets (total RTP, total hit rate, total volatility) are addressed in Sections 5–8.

3.1 Layer 1: Hit Rate via Window Coverage

For the standard left-anchored OAK ladder on R reels, the hit-rate frequencies are determined entirely by the window coverages $p_1, \dots, p_R$.

Proposition 3.1 (Hit-Rate Equations)**.** *The probability of an exact kOAK of symbol s is:*

$$H_k^{(s)} = \prod_{i=1}^{k} p_i^{(s)} \cdot (1 - p_{k+1}^{(s)}), \quad k = 2, \dots, R-1, \tag{9}$$

$$H_R^{(s)} = \prod_{i=1}^{R} p_i^{(s)}. \tag{10}$$

Hit rates are functions of p alone.

Proposition 3.2 (Suffix-Sum Solution). *Define* $S_k = \sum_{j \geq k} H_j = \prod_{i=1}^{k} p_i$ *(the probability that the chain reaches at least reel* k*). Then* $p_k = S_k/S_{k-1}$ *for* $k \geq 3$*, and* $p_1 \cdot p_2 = S_2$*, admitting one degree of freedom. This degree of freedom does not affect any OAK hit rate (since* H_k *for* $k \geq 2$ *depends on* $p_1 p_2 = S_2$*, not on* p_1 *and* p_2 *individually). It controls the* near-miss rate*: the probability that symbol* s *appears on reel 1 but not reel 2 is* $H_1^{(s)} = p_1(1-p_2) = p_1 - S_2$*, which ranges over* $[0, 1-S_2]$ *as* p_1 *ranges from* S_2 *to* 1*. The split is therefore a tease-frequency control with no effect on any paying outcome. Notably, the framework makes near-miss rates* auditable*: they are explicit design parameters with known values, not emergent properties of a heuristic strip. This transparency is relevant to responsible-gambling considerations, where near-miss frequency is a documented design concern.*

Remark 3.3 (Generalized Events). The OAK ladder is one event family. Any R-tuple of tri-state constraints (symbol present / symbol absent / unconstrained) defines an event whose hit rate factors as $\prod_{i \in \mathrm{SYM}} p_i \cdot \prod_{i \in \mathrm{NOT}} (1 - p_i)$. Near-miss patterns and non-contiguous events are handled identically.

3.2 Layer 2: RTP via Conditional Count

Two reels with identical p produce identical hit rates regardless of internal arrangement. They differ in RTP through c.

Theorem 3.4 (Independent RTP Control). *For fixed window coverages* $p_1, \ldots, p_R$*, the RTP (7) can be varied continuously by changing the stop counts* $n_1, \ldots, n_R$ *(and hence* $m_i = Wn_i/L_i$*) without altering any hit rate.*

Proof. Increasing n_i at fixed p_i increases $m_i = p_i \cdot c_i$ (since $c_i = Wn_i/a_i$ increases with n_i at fixed a_i), which increases the RTP contribution of every paying rung involving reel i via (7). Since hit rates (9)–(10) depend only on p, they are unaffected. □

Proposition 3.5 (Attainable Conditional Count). *For a symbol with* n *stops on a reel of length* L *with display height* W*, the attainable conditional counts from contiguous-block placement are:*

$$c_k = \frac{Wn}{n + k(W-1)}, \qquad k = 1, 2, \ldots, n, \tag{11}$$

where k *is the number of contiguous blocks, each separated by at least* $W - 1$ *other symbols. Each block of* n_j *stops covers* $n_j + W - 1$ *windows (the block plus two endcaps of* $W - 1$ *windows, which overlap with the block itself). Total coverage is* $a = n + k(W-1)$*. At* $k = 1$ *(one contiguous block),* c *is maximized at* $Wn/(n + W - 1)$*, approaching* W *as* $n \to \infty$*. At* $k = n$ *(all stops isolated),* $a = nW$ *and* $c = 1$*. The attainable set is dense in* $[1, W)$ *as* $L \to \infty$*, since for any rational target* $c^* = p/q \in [1, W)$*, a choice of* n *making* Wn/c^* *an integer achieves* c^* *exactly.*

Corollary 3.6 (Attainable RTP Range at Fixed Hit Rate). *At fixed window coverages* $p_1, \ldots, p_R$ *(fixing all hit rates), the RTP contribution of a* k*OAK of symbol* s *ranges over:*

$$v_k \cdot (1 - p_{k+1}) \cdot \prod_{i=1}^{k} p_i \;\leq\; \mathrm{RTP}_k^{(s)} \;\leq\; v_k \cdot (1 - p_{k+1}) \cdot \prod_{i=1}^{k} p_i \cdot c_{\max,i}, \tag{12}$$

where $c_{\max,i} = Wn_i/(n_i + W - 1)$ *is the maximum attainable conditional count on reel* i*. The lower bound corresponds to fully spread placement (*$c = 1$*,* $m = p$*). The upper bound corresponds to fully contiguous placement (*$c = c_{\max}$*,* $m = p \cdot c_{\max}$*). This range is computable before construction and provides a feasibility check: a target per-symbol RTP at a given hit rate is achievable if and only if it falls within these bounds.*

3.3 Layer 3: Volatility via Count Distribution

Definition 3.7 (Conditional Count Distribution)**.** For symbol s on reel i, let $C_i^{(s)}$ denote the random variable giving the count of s in a randomly sampled window, conditioned on the window being non-empty. The distribution of C lives on $\{1, 2, \ldots, W\}$.

Proposition 3.8 (Count Distribution at $W = 3$)**.** *At $W = 3$, the distribution of C on $\{1, 2, 3\}$ is determined by (c, τ^2) via:*

$$P(C = 3) = \tfrac{1}{2}(\tau^2 + c^2 - 3c + 2), \tag{13}$$
$$P(C = 2) = (c - 1) - 2P(C = 3), \tag{14}$$
$$P(C = 1) = 1 - P(C = 2) - P(C = 3). \tag{15}$$

The variance is bounded:

$$\tau^2_{\min} = (c - \lfloor c \rfloor)(\lceil c \rceil - c), \tag{16}$$
$$\tau^2_{\max} = (c - 1)(W - c). \tag{17}$$

Proof. Three unknowns (P_1, P_2, P_3) subject to three linear constraints: $\sum P_j = 1$, $\sum jP_j = c$, $\sum j^2 P_j = \tau^2 + c^2$. The system is full-rank; solving gives the stated formulas. The variance bounds follow from the support constraint $P_j \geq 0$: minimum variance concentrates mass at the integers nearest c; maximum variance places mass at $\{1, W\}$ only. □

Proposition 3.9 (Attainable Payout Variance)**.** *For a kOAK event of symbol s with pay value v_k, the expected payout is:*

$$\mathbb{E}[Y_k] = v_k \cdot (1 - p_{k+1}) \cdot \prod_{i=1}^{k} m_i, \tag{18}$$

which depends only on first moments (m_i and p_{k+1}). The second moment of the payout is:

$$\mathbb{E}[Y_k^2] = v_k^2 \cdot (1 - p_{k+1}) \cdot \prod_{i=1}^{k} m_{2,i}, \tag{19}$$

where $m_{2,i} = p_i \cdot (\tau_i^2 + c_i^2)$ is the unconditional second moment of the count on reel i. Since reels are independent, the product of second moments factors across reels. The payout variance is:

$$\mathrm{Var}(Y_k) = \mathbb{E}[Y_k^2] - \mathbb{E}[Y_k]^2. \tag{20}$$

At fixed p and c (fixing hit rate and RTP), the per-reel τ_i^2 takes values in a finite, computable set determined by the gap-pattern compositions of the cluster, contained within the moment-problem envelope $[\tau^2_{\min}, \tau^2_{\max}]$ (Proposition 3.8). Each realizable τ_i^2 gives a distinct per-event payout variance. The designer selects the payout volatility within this interval as an independent third specification: it does not affect the hit rate (a function of p alone) or the RTP (a function of $m = p \cdot c$ alone).

Remark 3.10 (Higher Display Heights)*.* At $W > 3$, the support $\{1, \ldots, W\}$ has more values than constraints, introducing additional degrees of freedom. This provides further design flexibility without affecting the independence result, which relies only on the fact that τ^2 does not enter the RTP formula.

Remark 3.11 (Scatter Scope). The first-order sufficiency of Proposition 2.11 holds for pay functions multilinear in per-reel counts, which includes the entire OAK ladder. Scatter symbols, which pay based on how many *reels* show the symbol (a sum, not a product), have pay functions nonlinear in the total count. For scatters, τ^2 enters the RTP. However, scatters in practice are placed as isolated singles ($c = 1$), in which case the per-reel count conditioned on presence is degenerate at 1 ($\tau^2 = 0$), and the scatter pays reduce to elementary symmetric functions of the per-reel presences p_i. Under this convention, first-order sufficiency is restored.

3.4 The Sequential Decoupling Theorem

Theorem 3.12 (Three-Layer Sequential Decoupling). *For pay functions multilinear in per-reel counts (the OAK ladder), the following three quantities can be specified independently per symbol per reel:*

1. ***Window coverage*** *p controls hit rate (Proposition 3.1). Coverage and conditional count are independently specifiable: the designer targets both p and c, and the construction derives $n = pcL/W$ and selects the cluster geometry accordingly (Proposition 3.5). The invariant $m = p \cdot c = Wn/L$ determines n; it does not couple p to c.*

2. ***Conditional count*** *c controls per-event RTP at fixed p (Theorem 3.4). Changing the number of stops n (and hence $c = Wn/a$) changes $m = p \cdot c$, which changes RTP, while p (and hence hit rate) remains unchanged.*

3. ***The conditional count distribution*** *controls per-event payout volatility at fixed p and c (Proposition 3.9). Different physical arrangements with the same p and c produce different distributions over $\{1, \ldots, W\}$, and hence different per-reel variances τ_i^2 and different payout variances. Since the RTP formula reads only $m_i = p_i \cdot c_i$ (a first-order moment) and the hit-rate formula reads only p_i, the count distribution is an independent third control.*

4 Existence and Constructibility

Sections 2 and 3 characterized the per-symbol, per-event design space: what can be specified, what ranges are attainable, and why the three layers are functionally decoupled. This section proves that any specification within those ranges can be realized by an actual reel strip.

4.1 Quasi-Contiguous Clusters and Symbol Independence

Definition 4.1 (Quasi-Contiguous Cluster). A *quasi-contiguous cluster* of symbol s is a sequence of n stops on the reel in which consecutive stops are separated by at most $W - 1$ positions. At this spacing, the window neighborhoods of consecutive stops overlap, so the cluster produces a contiguous range of non-empty windows. The internal gaps between consecutive stops may be occupied by other symbols.

Proposition 4.2 (Cluster Coverage and Conditional Count). *A quasi-contiguous cluster with span S (the distance from first stop to last) produces a contiguous range of $a = S+W$ non-empty windows, extending $W - 1$ positions beyond each end. The window coverage is $p = a/L = (S + W)/L$, fixed by the span and the reel length. The conditional count is:*

$$c = \frac{Wn}{S + W}, \tag{21}$$

which varies with the number of stops n packed into the span. More stops at fixed span increases c; fewer decreases it. The cluster contains $S+1-n$ internal gap positions available for other symbols.

The characterization is uniform: p is determined by the span, c by the ratio of stops to coverage, and the number of internal gaps by the difference. The fully contiguous case (gaps = 1, span = $n-1$) gives $a = n + W - 1$ and zero gap positions. The maximally spread case (gaps = $W - 1$, span = $(n-1)(W-1)$) gives the lowest c and the most gap positions.

Definition 4.3 (Separation)**.** Two clusters on the same reel are *separated* if at least $W-1$ positions not belonging to either cluster lie between them. Under separation, no window of height W contains stops from both clusters.

Proposition 4.4 (Symbol Independence)**.** *The count of symbol s in any window depends only on s's own stop positions. The statistics p_s, c_s, and τ_s^2 are therefore unconditionally decoupled across symbols: each is determined by its own stops alone, regardless of whether clusters are separated.*

Proof. The count of s in window $\{j, j+1, \ldots, j+W-1\}$ is the number of those positions occupied by stops of s. This is a function of s's positions alone. Other symbols' stops do not affect s's count in any window. □

Remark 4.5 (Role of Separation). Separation is a *design convenience*, not a mathematical necessity for symbol independence. It ensures $Q[s][t] = 0$ for separated designed pairs, simplifying the co-location analysis. Dropping separation tightens the minimum reel length bound (22) to $L \geq \sum n_s$ and returns the designed-designed block of Q as an additional design handle. The framework supports both separated and non-separated designs.

Proposition 4.6 (Interleaving)**.** *Let symbol A be placed in a quasi-contiguous cluster with span S_A, stop count n_A, and $G = S_A+1-n_A$ internal gap positions. A guest symbol B placed exclusively in A's gap positions is characterized by the same mathematics:*

1. *B's available positions are the G gap positions. B's maximum stop count is $n_B \leq G$.*
2. *B's coverage is a subset of A's coverage range. No additional reel space is consumed; B lives entirely within A's footprint.*
3. *B's conditional count is determined by B's own span within the gap positions: $c_B = Wn_B/(S_B+W)$, where S_B is B's span within the gaps. The same span-determines-p, stops-determine-c characterization applies.*
4. *A's statistics are unaffected by B's presence: they occupy disjoint positions and A's count in any window depends only on A's own stops (Proposition 4.4).*

The characterization is recursive: B's internal gaps can host a third symbol C, subject to the same mathematics. The separation cost ($W-1$ positions) is paid once for the outermost cluster, not per symbol.

4.2 Reel Length as a Derived Quantity

The reel length L is not a design input. It is derived from the specification.

Proposition 4.7 (Maximum Symbol Percent)**.** *A symbol with target (p, c) on a reel of length L occupies exactly $n = pcL/W$ positions, giving a symbol percent of $\mathrm{sp} = pc/W = m/W$. At the maximum attainable c (one contiguous block, $c_{\max} = Wn/(n+W-1) < W$), the symbol percent is strictly less than p. The footprint of a single contiguous block is n positions; with k blocks requiring $k(W-1)$ separation positions, the total footprint is $n + k(W-1)$.*

Each symbol s with k_s quasi-contiguous clusters, totaling n_s stops, requires n_s positions for its stops and at least $k_s(W-1)$ positions of intervening symbols to ensure separation between clusters. Without interleaving, the minimum reel length is:

$$L \geq \sum_{s \in \Omega} n_s + K \cdot (W-1), \tag{22}$$

where K is the total number of clusters across all symbols. With interleaving (Proposition 4.6), the bound is tighter: guest symbols occupy positions that count toward the host's gaps, reducing total footprint. Additionally, the total symbol percent of designed symbols must leave room for fillers:

$$\sum_{s \in \Omega_{\text{designed}}} \text{sp}^{(s)} = \sum_{s \in \Omega_{\text{designed}}} \frac{n_s}{L} < 1. \tag{23}$$

Since virtual reel strips have no physical length constraint [1], L can be chosen as large as needed to satisfy both bounds.

Proposition 4.8 (Integrality via LCM)**.** *Realizing a target (p, c) for symbol s requires $a = pL$ and $n = pcL/W$ to be positive integers. For a set of symbols with rational targets, let D be the least common multiple of all required denominators. Then $L = D$ (or any multiple of D) simultaneously satisfies all integrality constraints.*

Proof. Each rational target $p^{(s)} = a_s/b_s$ and $c^{(s)} = e_s/f_s$ requires L to be a multiple of b_s (so that $a = p^{(s)}L$ is an integer) and a multiple of $b_s f_s / \gcd(W, e_s b_s)$ (so that $n = p^{(s)}c^{(s)}L/W$ is an integer). Taking $D = \text{lcm}$ of all such denominators across all symbols gives the result. □

4.3 The Existence Theorem

Theorem 4.9 (Existence and Constructibility of the Skeleton)**.** *Let each symbol $s \in \Omega$ be assigned per-reel targets $(p_i^{(s)}, c_i^{(s)})$ with $c_i^{(s)} \in [1, W)$. Then for each reel i, a reel length L_i and a placement of all symbols exist such that every target is realized exactly. The construction is deterministic.*

Proof. Choose L_i satisfying the integrality constraints (Proposition 4.8) and the footprint bound (22). For each symbol s, the target $c_i^{(s)}$ determines the cluster geometry: by Definition 4.1, quasi-contiguous clusters achieve any integer coverage $a \in [n_s + W - 1, n_s W]$, so $c = Wn_s/a$ takes every value in a dense set. Place the clusters of each symbol on the reel, with at least $W-1$ positions of other symbols between any two clusters. By Proposition 4.4, the placements do not interfere: each symbol's (p, c) is realized by its own cluster structure. The remaining positions are filled by filler symbols (addressed in Section 5). □

Remark 4.10 (Lattice Resolution). Targets are realized exactly when they fall on the lattice $\{k/L : k \in \mathbb{Z}\}$. For targets off the lattice, the construction achieves the nearest lattice point, with granularity $O(1/L)$. Since L is a derived quantity with no upper bound, the designer may choose L as large as needed: longer reels provide finer-grained target resolution at no cost beyond strip length.

Proposition 4.11 (Reel Concatenation)**.** *Concatenating a strip with itself K times produces a strip of length KL that preserves all game metrics exactly: p, c, τ^2, m, $Q[s][t]/L$, RTP, hit rate, and variance are all invariant. This holds because the strip is cyclic: at the join between consecutive copies, the last $W-1$ symbols of one copy are identical to the first $W-1$ symbols of the next (they are the same strip). Therefore the windows spanning the join are the same windows that appear at the cyclic wrap in the original strip. No new window types are created; the multiset of windows is* *exactly K copies of the original multiset. The operation multiplies the number of stop positions by K, providing K-fold finer granularity for any quantity that depends on individual stop assignments (filler counts, value-bearing symbol values, gap orderings). Since concatenation is always available and introduces no side effects, the designer may assume arbitrarily fine resolution at every stage of the construction pipeline.*

Proof. Let $\mathbf{r} = (r_0, \ldots, r_{L-1})$ be the original cyclic strip and $\mathbf{r}^K$ the K-fold concatenation of length KL. For any position j in $\mathbf{r}^K$, the window $(r^K_j, r^K_{j+1}, \ldots, r^K_{j+W-1})$ equals $(r_{j \bmod L}, r_{(j+1) \bmod L}, \ldots, r_{(j+W-1) \bmod L})$ because the strip repeats with period L. Therefore the multiset of windows in $\mathbf{r}^K$ is exactly K copies of the multiset in $\mathbf{r}$. Every quantity computed as a ratio (count$/KL$, Q entry$/KL$, etc.) equals the corresponding ratio in the original (count$/L$, Q entry$/L$), since both numerator and denominator scale by K. □

Remark 4.12 (What This Theorem Does Not Address). Theorem 4.9 constructs a *skeleton*: each designed symbol placed with its target (p, c). It does not yet address total game RTP (which depends on the filler composition, Section 5), total game hit rate (which depends on cross-symbol co-occurrence, Sections 6–7), or total game volatility (Section 8). These are handled by subsequent sections, each building on the skeleton without modifying it.

5 Filler System and Total RTP

The skeleton from Section 4 places every designed symbol with its target (p, c). Every remaining position on the reel is occupied by a *filler symbol*: a paying symbol whose count is determined by the RTP allocation and whose clustering is a design parameter controlling hit rate.

5.1 Filler Placement

Definition 5.1 (Filler Symbol)**.** A *filler symbol* $f \in \Omega$ is a paying symbol whose placement is not fixed by the skeleton. On blocker reels (reel $d_f + 1$ for a depth-d_f symbol), $c_f = 1$ (singles): no two stops within $W - 1$ positions, so $p_f = m_f = Wn_f/L$ is determined by counts alone. On non-blocker reels $(1, \ldots, d_f)$, $c_f \geq 1$ is a design parameter: clustering reduces p_f at fixed m_f (Theorem 2.7), which is the primary hit-rate control (Corollary 2.13).

Every position not occupied by a designed symbol or a wild is assigned to a filler. No reel position is left blank. The total number of filler positions on reel i is $L_i - \sum_{s \in \Omega_{\text{designed}}} n_s^{(i)}$. These positions are distributed among F filler symbols, each with its own pay values in the paytable.

5.2 Filler RTP Formula

The RTP formula reads $m_i = Wn_f^{(i)}/L_i$ on paying reels (arrangement-invariant) and p_i on blocker reels. On blocker reels, fillers are singles ($c_f = 1$), so $p_f = m_f = Wn_f/L$ — determined by counts alone. On non-blocker reels, p_f does not enter the RTP formula (there is no blocker term). Therefore the filler RTP is a pure function of counts regardless of clustering:

$$\mathrm{RTP}_k^{(f)} = v_k^{(f)} \cdot \prod_{i=1}^{k} \frac{Wn_f^{(i)}}{L_i} \cdot \left(1 - \frac{Wn_f^{(k+1)}}{L_{k+1}}\right), \quad k < R. \tag{24}$$

No arrangement parameters appear. The total filler RTP is:

$$\mathrm{RTP}_{\text{filler}} = \sum_f \sum_{k=2}^{R} \mathrm{RTP}_k^{(f)}. \tag{25}$$

The total game RTP is the sum of the skeleton contribution (fixed by Section 4) and the filler contribution:

$$\mathrm{RTP}_{\text{total}} = \mathrm{RTP}_{\text{skeleton}} + \mathrm{RTP}_{\text{filler}}. \tag{26}$$

5.3 Multilinearity of Filler RTP

Lemma 5.2 (Multilinearity)**.** *The total filler RTP is multilinear in the per-reel filler count vectors: with all other reels held fixed, RTP is an affine function of any single reel's filler counts.*

Proof. In (24), each OAK rung involves each reel at most once: reel i contributes either $Wn_f^{(i)}/L_i$ (linear in $n_f^{(i)}$) as a paying reel, or $1 - Wn_f^{(i)}/L_i$ (affine in $n_f^{(i)}$) as a blocker. No rung contains a product of two quantities from the same reel. The total filler RTP, a sum of such rungs, is therefore affine in each reel's filler counts with all others fixed. □

Remark 5.3. Multilinearity has a practical consequence beyond finding extrema. It means the *marginal RTP contribution* of each filler on each reel is a constant (at fixed counts on other reels). Moving one stop of filler f from reel i to reel j changes the RTP by a predictable, pre-computable amount. Each candidate reallocation can be scored in $O(1)$ time from a pre-computed marginal table, rather than requiring a full RTP recomputation.

5.4 RTP Band

Corollary 5.4 (Extrema at Vertices)**.** *Filler RTP is multilinear across reels (Lemma 5.2): affine in each reel's count vector with the others held fixed [12]. Over the box-constrained simplex $\sum_f n_f^{(i)} = N_i$, $0 \le n_f^{(i)} \le \lfloor L_i/W \rfloor$, an affine function's extrema are attained at vertices [15]. Since reel-by-reel optimization preserves this property (each reel's optimum is at a vertex of its own box-constrained simplex, regardless of the others' positions), the global extrema are attained at vertices of the product polytope.*

Definition 5.5 (RTP Band)**.** The *RTP band* is the interval $[\mathrm{RTP}_{\min}, \mathrm{RTP}_{\max}]$ of total game RTP values achievable by varying the filler composition within the box-constrained simplex, with all designed-symbol placements fixed.

Proposition 5.6 (RTP Band Width)**.** *The RTP band has positive width whenever at least two filler symbols have distinct pay values. The width depends on the spread of filler pay values and the proportion of the reel occupied by fillers. A larger filler region (lower skeleton footprint) and a wider spread of filler pay values produce a wider band.*

Proof. If two fillers f_1, f_2 have distinct pay values, then moving one stop from f_1 to f_2 on some reel changes the filler RTP (since the marginal RTP contribution of f_1 and f_2 differ on at least one rung). The floor and ceiling vertices therefore differ, and the band has positive width. □

Remark 5.7 (Feasibility Check)*.* The RTP band provides a pre-construction feasibility check. Before any filler is placed, the designer computes the band from the skeleton and the paytable. If the target total RTP falls outside the band, the skeleton must be adjusted (by modifying designed symbol counts) before construction can proceed. If it falls inside, the filler system is guaranteed to reach it.

5.5 Total RTP Targeting

Proposition 5.8 (Total RTP Is Achievable)**.** *Any total RTP within the RTP band is achievable to within the lattice resolution* $\delta(L) = O(1/L)$.

Proof. The filler allocation has $R(F-1)$ continuous degrees of freedom (each of R reels distributes its filler budget among F filler symbols, subject to one sum constraint per reel). The feasible set of allocations forms a convex polytope $\mathcal{P}$ (the Cartesian product of R simplices).

Let $\mathbf{x}_{\min}, \mathbf{x}_{\max} \in \mathcal{P}$ be the vertex allocations achieving $\mathrm{RTP}_{\min}$ and $\mathrm{RTP}_{\max}$ respectively (Corollary 5.4). Since $\mathcal{P}$ is convex, the path

$$\mathbf{x}(t) = (1-t)\,\mathbf{x}_{\min} + t\,\mathbf{x}_{\max}, \qquad t \in [0,1], \tag{27}$$

lies entirely within $\mathcal{P}$. The filler RTP along this path is a continuous function $\mathrm{RTP}(t)$ of one real variable, with $\mathrm{RTP}(0) = \mathrm{RTP}_{\min}$ and $\mathrm{RTP}(1) = \mathrm{RTP}_{\max}$. By the intermediate value theorem, for every target $\mathrm{RTP}^* \in [\mathrm{RTP}_{\min}, \mathrm{RTP}_{\max}]$, there exists $t^* \in [0,1]$ such that $\mathrm{RTP}(t^*) = \mathrm{RTP}^*$.

The continuous allocation $\mathbf{x}(t^*)$ may have non-integer counts. Rounding via the largest-remainder method (floor all counts, then increment those with the largest fractional parts until $\sum n_f = N_i$) preserves the budget constraint exactly and changes each count by at most 1. Each unit change shifts the total RTP by $v_k \cdot k \cdot m^{k-1} \cdot (W/L) \cdot (1 - p_{k+1})$ per symbol per OAK level. The aggregate resolution $\delta(L)$ depends on the paytable; since L is a derived quantity with no upper bound (Proposition 4.11), the resolution can be made arbitrarily fine. □

5.6 Side Effects and Separation of Concerns

Remark 5.9 (Filler Allocation and Hit Rate)*.* Changing filler counts changes filler window coverages ($p_f = Wn_f/L$), which changes the total game hit rate. The filler allocation is not chosen for RTP alone: it is selected as the operating point on the RTP iso-surface where the target hit rate falls within the co-location-achievable range and the target volatility falls within the clustering-achievable range. The joint selection is formalized in the Full Construction Theorem (Section 9).

Remark 5.10 (Skeleton Preservation)*.* The filler allocation modifies only filler counts. The designed symbols' stop counts, placements, coverages, conditional counts, and count distributions are unchanged. Every per-symbol target established by the skeleton (Section 4) is preserved exactly.

Remark 5.11 (Pre-Computability)*.* The RTP band and the per-reallocation RTP change are computable from the specification and the skeleton before any filler is placed. The designer knows, before construction begins, whether a target total RTP is achievable alongside the per-symbol targets.

6 The Co-location Matrix

Sections 4 and 5 construct a skeleton and target total RTP. The remaining global targets are total hit rate and total volatility, both of which depend on which symbols share windows. This section introduces the mathematical object that captures that structure.

6.1 Definition and Basic Properties

Definition 6.1 (Co-location Matrix)**.** For a reel strip $\mathbf{r}$ of length L with display height W, the *co-location matrix* Q is the $|\Omega| \times |\Omega|$ symmetric matrix defined by:

$$Q[s][s] = \#\{j : s \in \text{window}(j)\}, \tag{28}$$

$$Q[s][t] = \#\{j : s \in \text{window}(j) \text{ and } t \in \text{window}(j)\}, \quad s \neq t. \tag{29}$$

The diagonal records per-symbol presence counts. The off-diagonal records pairwise co-occurrence counts. Q is the complete order-2 window statistic of the strip.

Proposition 6.2 (Margin Laws)**.** *Under the singles constraint (no symbol appears twice in any window), the following identities hold:*

$$Q[s][s] = W \cdot n_s, \tag{30}$$

$$\sum_{t \neq s} Q[s][t] = (W-1) \cdot Q[s][s], \tag{31}$$

$$\sum_{s<t} Q[s][t] = \binom{W}{2} \cdot L. \tag{32}$$

Proof. (30): each of the n_s stops of s appears in exactly W windows. Under singles, no window counts s twice, so $Q[s][s] = Wn_s$.

(31): each window containing s contains exactly $W-1$ other symbols (under singles). Summing over all $Q[s][s]$ such windows gives the row sum.

(32): each of the L windows contributes $\binom{W}{2}$ to the total off-diagonal mass (one for each pair of distinct symbols in the window). □

Corollary 6.3 (Diagonal Is Fixed by Counts)**.** *The diagonal of Q is entirely determined by the symbol counts $\{n_s\}$, which are fixed by the skeleton and the filler allocation. It carries no design freedom.*

6.2 Windows as Triangles

Proposition 6.4 (Windows as Triangles)**.** *Under singles at $W = 3$, each window contains exactly 3 distinct symbols $\{a, b, c\}$ and contributes $+1$ to each of $Q[a][b]$, $Q[a][c]$, and $Q[b][c]$. A window is therefore a* triangle *on the symbol graph weighted by Q. A strip of length L induces a decomposition of the multigraph with edge weights $Q[s][t]$ into exactly L triangles. Conversely, a matrix Q satisfying the margin laws is realizable by a strip only if the corresponding multigraph admits such a decomposition.*

Remark 6.5 (Generalization to $W > 3$). At display height W, each window is a clique of size W, contributing $+1$ to $\binom{W}{2}$ off-diagonal entries. The triangle decomposition generalizes to a W-clique decomposition. The results of this section hold at any W; the triangle language is used for concreteness at $W = 3$.

6.3 Design Freedom

Proposition 6.6 (Q Decomposition Under a Skeleton)**.** *When a skeleton is present (designed symbols at fixed positions), the co-location matrix decomposes into three blocks:*

1. Skeleton-skeleton*: entries $Q[s][t]$ where both s and t are designed symbols. These are fully determined by the skeleton geometry and cannot be changed.*

2. Skeleton-filler*: entries $Q[s][f]$ where s is a designed symbol and f is a filler. The skeleton determines which windows contain s; the filler arrangement determines which filler occupies the remaining cells in those windows. These entries are partially constrained by the skeleton but admit design freedom in the filler assignment.*

3. Filler-filler*: entries $Q[f_1][f_2]$ where both are fillers. These are the main design space. Under singles (which holds for fillers on blocker reels), the margin laws apply exactly in this block.*

The skeleton contribution to the total Q is pre-computable in $O(L)$ from the skeleton geometry alone. The filler contribution is the design freedom. The total is additive: $Q_{\text{total}} = Q_{\text{skeleton}} + Q_{\text{filler}}$.

Remark 6.7 (Degenerate Windows). When a designed symbol has $c > 1$, some windows contain multiple instances of that symbol, violating the singles constraint locally. These *degenerate windows* affect the margin laws for both skeleton and filler entries: a filler symbol appearing in a degenerate window has fewer than $W - 1$ distinct neighbors, reducing its row sum (see Proposition 10.6 for the explicit correction). The degenerate window count and the per-filler deficit d_s are computable in advance from the skeleton geometry. By Corollary 2.13, filler clustering is restricted to non-blocker reels, so degenerate filler windows occur only on reels where p does not enter the RTP formula. On blocker reels, fillers remain as singles.

Proposition 6.8 (Off-Diagonal Freedom)**.** *The number of free parameters in Q is $\binom{n}{2} - n$, where $n = |\Omega|$. The diagonal is fixed by counts (30), and the row-sum law (31) imposes n linear constraints on the $\binom{n}{2}$ off-diagonal entries.*

Proof. There are $\binom{n}{2}$ off-diagonal entries (upper triangle). Each of the n row-sum constraints fixes one linear combination of the off-diagonal entries in that row. The total mass constraint (32) is implied by summing the row-sum constraints, so it is not independent. The remaining freedom is $\binom{n}{2} - n$. □

Remark 6.9. For $n = 8$ filler symbols, the freedom is $28 - 8 = 20$ dimensions. For $n = 10$, it is $45 - 10 = 35$. This is the space within which hit rate and volatility can be moved at frozen RTP.

6.4 RTP Invisibility

Proposition 6.10 (Q Cannot Move RTP)**.** *Any change to the off-diagonal block of Q (at fixed diagonal) preserves RTP exactly.*

Proof. The RTP formula (7) reads each reel through $m_i = Wn_i/L$ (on paying reels, invariant by Theorem 2.7) and $p_i = a_i/L$ (on blocker reels). For filler symbols on blocker reels (singles), $p_i = Wn_i/L = m_i$, determined by counts alone. For designed symbols, p_i is fixed by the skeleton (which is not modified). In both cases, the quantities RTP reads are determined by either the counts or the skeleton, neither of which changes when the filler arrangement changes. The off-diagonal entries of Q, which record pairwise co-occurrence, do not appear in the RTP formula. Therefore any rearrangement of fillers that preserves counts preserves RTP exactly. □

Remark 6.11. This is the structural foundation of the co-location approach: the off-diagonal block of Q is the design space that moves hit rate and payout variance while being invisible to RTP. The asymmetry arises because RTP is multilinear in per-reel counts (Proposition 2.11), and pairwise co-occurrence is an order-2 quantity that does not enter a multilinear formula.

6.5 The Transposition Lattice

Definition 6.12 (Transposition)**.** A *transposition* exchanges the symbols at two positions on the strip. It preserves all symbol counts and hence all diagonals and all row sums.

Proposition 6.13 (Lattice Rank)**.** *The lattice of achievable Q-differences generated by transpositions has rank exactly $\binom{n}{2} - n$, equal to the off-diagonal freedom (Proposition 6.8). No hidden linear invariant obstructs movement: any achievable difference is an integer combination of transposition deltas.*

Proof. A rotation on quadruple (a, b, c, e) changes four entries: $Q[a][b]$ and $Q[c][e]$ by $+\delta$, $Q[a][e]$ and $Q[c][b]$ by $-\delta$. This difference vector lies in the integer kernel of the unsigned vertex-edge incidence matrix B of K_n (the row-sum constraints $\sum_{t \neq s} Q[s][t] = r_s$ are exactly $B\mathbf{q} = \mathbf{r}$, so any difference in $\ker_{\mathbb{Z}} B$ preserves all row sums). For non-bipartite K_n ($n \geq 3$), the rank of B over $\mathbb{Q}$ is n (Proposition 6.8), so $\dim(\ker_{\mathbb{Q}} B) = \binom{n}{2} - n$. The rotation generators (alternating 4-cycles in K_n) span $\ker_{\mathbb{Z}} B$: any kernel vector can be reduced to zero by subtracting generators, using completeness of K_n to route any edge-pair through a 4-cycle. At $n = 4$: $\ker_{\mathbb{Z}} B$ has rank $\binom{4}{2} - 4 = 2$; the three alternating 4-cycles satisfy one linear dependence, spanning the full kernel. For general $n \geq 3$: any kernel vector is reduced to zero by iteratively subtracting 4-cycle generators, using completeness of K_n to route each nonzero edge-pair through a common vertex. □

Remark 6.14. The lattice rank establishes that transpositions span the full design freedom. It does not establish which specific Q matrices are achievable, only that the achievable set has the expected dimension.

6.6 Row-Sum-Preserving Rotations

Definition 6.15 (Rotation)**.** A *row-sum-preserving rotation* on Q selects four distinct symbols a, b, c, e and an integer δ, and applies:

$$Q[a][b] \leftarrow Q[a][b] + \delta, \qquad Q[c][e] \leftarrow Q[c][e] + \delta, \tag{33}$$

$$Q[a][e] \leftarrow Q[a][e] - \delta, \qquad Q[c][b] \leftarrow Q[c][b] - \delta. \tag{34}$$

The rotation preserves all row sums, all diagonals, all symbol counts, and hence all RTP quantities. It is *valid* if all four resulting entries remain non-negative. Every transposition induces a rotation (or a sum of rotations) on Q.

Remark 6.16. Rotations are the atomic operations in the off-diagonal design space. Any two Q matrices with identical margins are connected by a sequence of rotations.

6.7 How Q Enters the Hit Rate

The total game hit rate depends on per-reel moments through the inclusion-exclusion principle.

Proposition 6.17 (Hit Rate via Inclusion-Exclusion)**.** *For a game with R identical reels and pay depth d (minimum OAK level), the probability of at least one win is:*

$$H = \sum_{k=1}^{W} (-1)^{k+1} \sum_{\substack{S \subseteq \Omega \\ |S| = k}} \prod_{i=1}^{d} \frac{U_i[S]}{L_i}, \tag{35}$$

where $U_i[S] = \#\{$*windows on reel* i *containing every symbol in* $S\}$*. The order-*1 *terms* $U[\{s\}] = Q[s][s]$ *are fixed by counts. The order-*2 *terms* $U[\{s,t\}] = Q[s][t]$ *are the off-diagonal entries of* Q*. Terms of order* 3 *and above are not determined by* Q*.*

Proof. The inclusion-exclusion over symbol subsets gives the probability that at least one symbol is present on all d paying reels. For a subset S, the probability that every symbol in S is present on reel i is $U_i[S]/L_i$. Independence of the reels gives the product across reels. Under singles, a window has at most W distinct symbols, so $U[S] = 0$ for $|S| > W$ and the sum terminates at $k = W$. □

Remark 6.18. The formula extends to per-reel Q_i (non-identical reels) by replacing $(Q[s][t]/L)^d$ with $\prod_{i=1}^{d} Q_i[s][t]/L_i$. For symbol-dependent pay depths d_s, the pairwise intersection term for symbols s (depth d_s) and t (depth d_t, $d_t > d_s$) uses Q entries on reels $1, \ldots, d_s$ and the deeper symbol's marginal presence on reels $d_s{+}1, \ldots, d_t$:

$$P(A_s \cap A_t) = \prod_{i=1}^{d_s} \frac{Q_i[s][t]}{L_i} \cdot \prod_{i=d_s+1}^{d_t} \frac{Q_i[t][t]}{L_i}.$$

The worked example has depths $d \in \{2, 3\}$, so this form is exercised directly.

Corollary 6.19 (Order-2 Hit Rate)**.** *The* order-2 hit rate *is the truncation of (35) at* $k = 2$*:*

$$H^{(2)} = \sum_{s \in \Omega} \left(\frac{Q[s][s]}{L} \right)^d - \sum_{s<t} \left(\frac{Q[s][t]}{L} \right)^d. \tag{36}$$

This is a function of Q *alone.*

6.8 Exact Order-2 Hit-Rate Invariant

Theorem 6.20 (Order-2 Hit-Rate Invariant Condition)**.** *At pay depth* $d = 3$*, a rotation* $(a, b) \leftrightarrow (c, e)$ *at integer* δ *preserves the order-2 hit rate* $H^{(2)}$ *exactly if and only if:*

$$\delta = -\frac{S_1 D_1 + S_2 D_2}{S_1 + S_2}, \tag{37}$$

where $S_1 = Q[a][b] + Q[a][e]$*,* $D_1 = Q[a][b] - Q[a][e]$*,* $S_2 = Q[c][e] + Q[c][b]$*,* $D_2 = Q[c][e] - Q[c][b]$*. The rotation is valid when* δ *is a nonzero integer and all four resulting entries remain non-negative.*

Proof. At pay depth $d = 3$, each pair's contribution to $H^{(2)}$ is $(Q[s][t]/L)^3$. The rotation changes four entries by $\pm\delta$. Expanding:

$$\Delta H^{(2)} = -\frac{3\delta}{L^3} \left[S_1 D_1 + S_2 D_2 + \delta(S_1 + S_2) \right], \tag{38}$$

since the cubic terms cancel: $\delta^3(1 + 1 - 1 - 1) = 0$. Setting $\Delta H^{(2)} = 0$ (with $\delta \neq 0$) gives (37). □

Corollary 6.21 (Consecutive-Entry Rotations Preserve Hit Rate)**.** *The special case* $\delta = 1$ *gives* $S_1(D_1 + 1) + S_2(D_2 + 1) = 0$*, satisfied when* $Q[a][e] = Q[a][b] + 1$ *and* $Q[c][b] = Q[c][e] + 1$ *(both factors vanish independently). Such rotations swap consecutive values* $(A, A{+}1) \to (A{+}1, A)$*, permuting the multiset of off-diagonal entries. Consequently* $\sum Q[s][t]^d$ *is preserved for* every d *simultaneously, making the family depth-universal.*

Non-unit solutions to (37) are depth-specific: they preserve $d = 3$ *but generally not* $d = 2$ *or* $d = 4$*. Only the consecutive family (*$\delta = 1$*) is safe for a total hit rate that sums across OAK depths. However, at fixed depth, the general formula provides far more available rotations: any four entries satisfying (37) at integer* δ *yield a valid move.*

Proof. Substituting $Q[a][e] = Q[a][b] + 1$ into $S_1(D_1 + 1)$: $S_1 \cdot 0 = 0$. Similarly for the second factor. Both terms vanish independently. Since the rotation swaps the two values in each pair, the multiset $\{Q[s][t] : s < t\}$ is unchanged, so $\sum Q[s][t]^d$ is invariant for all d. □

Remark 6.22. Symmetric rotations (equal entries spread apart) do *not* preserve hit rate: x^d is strictly convex, so $(A + 1)^d + (A - 1)^d > 2A^d$ for $d \geq 2$. The hit-rate-preserving rotations are those between asymmetric entries where the convex increase from the smaller entries is exactly compensated by the concave decrease from the larger entries. The consecutive-entry condition provides a constructive family of such rotations. With $\binom{n}{2} - n$ off-diagonal degrees of freedom and hit rate being one scalar, the null space of hit-rate-preserving moves has dimension at least $\binom{n}{2} - n - 1$. This family of moves is available for volatility fine-tuning (Section 8), though count allocation and clustering provide the dominant variance controls.

6.9 How Q Enters Payout Variance

Proposition 6.23 (Per-Reel Payout Second Moment and Q)**.** *For analysis of cross-symbol co-occurrence effects, consider the per-reel linear surrogate $X = \sum_s v_s \cdot C_s$, where C_s is the count of symbol s in the window. This is not the ways-pay payout (which involves products across reels) but captures the per-reel contribution to variance. The second moment is:*

$$\mathbb{E}[X^2] = \sum_s v_s^2 \cdot \frac{Q[s][s]}{L} \cdot (\tau_s^2 + c_s^2) + 2 \sum_{s<t} v_s v_t \cdot \frac{Q[s][t]}{L}, \tag{39}$$

where the first sum involves the diagonal (per-symbol second moments, from Layer 3) and the second sum involves the off-diagonal (pairwise co-occurrence). The cross terms are exactly the off-diagonal entries of Q, scaled by the product of pay values.

Proof. $\mathbb{E}[X^2] = \sum_{s,t} v_s v_t \mathbb{E}[C_s C_t]$. For $s = t$: $\mathbb{E}[C_s^2] = (Q[s][s]/L)(\tau_s^2 + c_s^2)$. For $s \neq t$ under singles (both $c_s = c_t = 1$): at most one of each symbol appears per cell, so if both are present, $C_s C_t = 1$. Thus $\mathbb{E}[C_s C_t] = P(\text{both present}) = Q[s][t]/L$. For designed symbols with $c > 1$, $\mathbb{E}[C_s C_t]$ depends on the joint count distribution within shared windows and is not simply $Q[s][t]/L$; the designed-designed cross terms are handled as fixed skeleton contributions (Proposition 6.6). □

Corollary 6.24 (Volatility Control via Off-Diagonal)**.** *At fixed counts (fixed diagonal) and fixed per-symbol distributions (fixed τ_s^2, c_s^2), the payout variance is an affine function of the off-diagonal entries of Q. Increasing $Q[H_1][H_2]$ (two high-pay symbols share windows more often) increases variance. The off-diagonal block is the handle for total payout volatility at frozen RTP and frozen per-event volatility.*

6.10 The Attainable Set

Not every matrix satisfying the margin laws is achievable. Realizability is constrained by integrality and the window-content structure.

Definition 6.25 (Feasibility Conditions)**.** A candidate matrix Q satisfying the margin laws is subject to progressively stronger feasibility conditions:

1. *Row sums*: Q satisfies (30)–(32).

2. *W-subset consistency*: for every subset S of W symbols, the pairwise entries $Q[s][t]$ for $s, t \in S$ must be jointly achievable by some set of triangles on S.

3. 3-*gram linear system*: there exists a non-negative integer assignment of counts to each of the $n(n-1)(n-2)$ ordered 3-grams satisfying flow conservation, symbol counts, and the Q constraints.

4. *Connectivity*: the support of the 3-gram assignment has connected support (is a single connected component).

Proposition 6.26 (Feasibility Ladder (Computational))**.** *Each condition is strictly stronger than the previous one. Exhaustive enumeration at* $n = 5$, $L = 10$ *yields:*

Condition	***Matrices admitted***
Row sums only	*10,577*
+ W*-subset consistency*	*158*
+ 3*-gram linear system*	*148*
+ *Connectivity*	***87***

The final condition admits exactly the achievable matrices. Soundness (achievable $\Rightarrow$ *passes all four) holds because any physical strip's* Q *satisfies all four conditions by construction. Completeness (passes all four* $\Rightarrow$ *achievable) follows from the Euler–Hierholzer theorem [11, 8]: at* $W = 3$, *windows are the* 3*-grams of the cyclic strip. A non-negative integer* 3*-gram count vector is realizable as a cyclic sequence if and only if it satisfies flow conservation on the* 2*-gram de Bruijn graph and has connected support — an Eulerian circuit on the directed multigraph. Conditions 3 (the* 3*-gram linear system) and 4 (connectivity) are precisely these two requirements. This holds at all* (n, L), *not only at the enumerated parameters. At general* W, *the same argument applies with* $(W-1)$*-gram nodes and* W*-gram edges.*

6.11 Constructibility from Q

The feasibility ladder (Proposition 6.26) operates at the 3-gram level. In practice, the designer holds Q — the pair matrix — not the full 3-gram census. The question is whether a given $(L, \mathbf{n}, Q)$ admits *some* strip realizing it, without specifying the census.

Proposition 6.27 (Constructibility Test)**.** *A specification* $(L, \mathbf{n}, Q)$ *is realizable by a cyclic strip if and only if there exists a non-negative integer* W*-gram count vector* $\mathbf{y}$ *satisfying: (1) the* Q *constraints (pairwise sums over* $\mathbf{y}$ *reproduce* Q*), (2) flow conservation on the* $(W-1)$*-gram de Bruijn graph, and (3) strong connectivity of the support. This is an integer feasibility problem over* n^W *variables and* n^{W-1} *nodes, with* L *appearing only in the right-hand sides. A YES returns* $\mathbf{y}$, *and Hierholzer's algorithm reads the strip off in* $O(L)$ *time.*

Remark 6.28 (The Gap Is Real). Comparing constraint-legal Q matrices (satisfying margin laws, integrality, and non-negativity) against those realizable by a strip, by exhaustive enumeration at small parameters:

n	L	Constraint-legal	Realizable	Not realizable
3	9	249	54	78.3%
3	12	606	261	56.9%
4	8	4489	157	96.5%

At $n = 4$, roughly 97 times in 100 a constraint-legal Q is unrealizable. The constructibility test (Proposition 6.27) classifies all 801 cases with zero false positives and zero false negatives, validating the implementation against exhaustive ground truth. The correctness of the test itself follows from the Euler–Hierholzer theorem (Proposition 6.26), which holds at all (n, L). Rejection costs ∼60 ms; acceptance with the built strip costs ∼0.4–2.6 s. Rejection is ∼20× cheaper than acceptance, which is the right asymmetry for a design loop that proposes many candidates.

Remark 6.29 (The Realizable Set Is Connected). Once on the realizable set, every Q with the same counts is reachable from every other via coordinated moves (two pair counts up, two down). The set is a single connected component under L_1-radius-4 moves, with diameter 4–8 hops, at $n = 3, 4, 5$ and $L = 8..12$. This means the design system can navigate the realizable set freely without leaving it, provided moves are coordinated rather than single-entry.

Remark 6.30 (Q-Space Design Loop). With constructibility decidable from Q in milliseconds, the design loop operates entirely in matrix space:

1. Evaluate RTP and volatility from $(L, \mathbf{n}, Q)$: $O(n^2)$, ∼10 ms.
2. Navigate Q via coordinated moves: $O(n^2)$ per step.
3. Check constructibility at each candidate: ∼60 ms (reject) or ∼2.6 s (accept).
4. Build the strip once, from the final Q: $O(L)$ via Hierholzer.

The strip is never touched during design. Q is the order-2 design object (complete for RTP-invisibility and the hit-rate oracle; the full design additionally carries the W-gram census). Full pipeline from specification to verified strips: ∼17 s.

6.12 Reachability by Mixing

The co-location matrix Q specifies the arrangement. This subsection shows that the achievable set of $(H, \mathrm{CV}_{\mathrm{win}})$ pairs at fixed counts is *reachable by mixing*, and that any interior point is achieved by splicing rather than searching.

Proposition 6.31 (Exact Moment Preservation under Concatenation)**.** *A cyclic strip concatenated with itself has identical normalized moments: m/L, $Q[s][t]/L$, p, c, and all higher-order statistics are bit-identical. Concatenation scales the integer lattice without changing any game metric.*

Proof. The K-fold concatenation of a strip of length L is a strip of length KL. Every window of the concatenated strip is a window of the original strip, and each original window appears exactly K times. Therefore every normalized statistic (computed as a sum over windows divided by total windows) is identical. □

Proposition 6.32 (Splice Linearity)**.** *Let A and B be two strips of equal length L sharing the same skeleton and the same per-symbol counts. The spliced strip $A^{n-k}B^k$ ($(n-k)$ copies of A followed by k copies of B) has moment vector:*

$$\mu(A^{n-k}B^k) = \frac{n-k}{n}\,\mu(A) \;+\; \frac{k}{n}\,\mu(B) \;+\; O(1/n), \tag{40}$$

where μ includes all per-reel presences, co-locations, and payout moments. The $O(1/n)$ term is a junction correction arising from the $W-1$ windows that straddle block boundaries. When the skeleton pins the first and last $W-1$ cells of each block, all homogeneous junctions ($A|A$ and $B|B$)

produce windows identical to the original strip's wrap, contributing zero correction. The single heterogeneous cyclic wrap (the last B block returning to the first A block) contributes at most $W-1$ windows out of nL total — a correction of $O(1/(nL))$, negligible at production lengths.

Proof. Each block contributes L windows (cyclic within the block) plus $W-1$ junction windows at each boundary. With n blocks there are n boundaries contributing $n(W-1)$ junction windows and $n(L-W+1)$ interior windows, totalling nL. The interior moments are the weighted average $((n-k)\mu(A)+k\mu(B))/n$ by linearity. The junction fraction per block is $(W-1)/L$, independent of n. The $O(1/n)$ term in (40) is the resolution of the mixing ratio k/n, not the junction size. When the skeleton pins the boundary cells, both A and B produce the same junction windows (e.g., both end with `H1 filler` and begin with `filler 10`), so the junction correction is zero. □

Corollary 6.33 (Reachability by Mixing)**.** *Let $\mathcal{S}$ be the set of strips sharing a given skeleton and count vector. The achievable set of moment vectors $\{(H(S),\mathbb{E}[Y^2](S)) : S \in \mathcal{S}\}$ is convex, since both H and $\mathbb{E}[Y^2]$ are affine along the splice path. Since $\mathrm{CV}^2_{\mathrm{win}} = H \cdot \mathbb{E}[Y^2]/\mathrm{RTP}^2 - 1$ is a product of two affine quantities at fixed RTP, the achievable $(H, \mathrm{CV}_{\mathrm{win}})$ pairs trace a* parabolic arc *under a two-block splice, not a segment. The arc bulges outward (decreasing H increases $\mathbb{E}[Y^2]$, so the quadratic term is negative), meaning the achievable set in $(H, \mathrm{CV}_{\mathrm{win}})$ is a smooth image of a convex set and the interior is reachable. Note: these statements hold for single-reel splices (one reel varied, others fixed). Since H is a product across reels, a simultaneous multi-reel splice at common ratio t produces a degree-R polynomial in t, not an affine function. Multi-reel targets are reached by sequential single-reel splices, each preserving the previous reel's statistics.*

Proof. For any two achievable points $\mu(A)$ and $\mu(B)$, Proposition 6.32 shows that every rational convex combination $\frac{k}{n}\mu(A) + \frac{n-k}{n}\mu(B)$ is achieved by the splice $A^{n-k}B^k$. Both H and $\mathbb{E}[Y^2]$ are sums over windows, hence affine in the splice ratio. The derived coordinate $\mathrm{CV}_{\mathrm{win}}$ is a smooth function of these affine quantities, so the achievable set in $(H, \mathrm{CV}_{\mathrm{win}})$ is a smooth curve. Three blocks suffice to cover a two-dimensional region (Proposition 6.35). Precision costs strip length, linearly. □

Remark 6.34 (Construction by Mixing)*.* Reachability by mixing reduces construction to three steps: (1) build extreme-point strips at the boundary of the achievable set, (2) choose the splice ratio that hits the target, (3) concatenate. No Monte Carlo, no fitness function, no convergence hope. The extreme points are found by maximizing and minimizing the hit rate over the filler arrangement freedom (the off-diagonal block of Q). The interior is reached by mixing. The RTP is preserved exactly throughout because all blocks share the same counts. The hit-rate resolution is $O(1/n)$; the $\mathrm{CV}_{\mathrm{win}}$ resolution is $O(1/n)$. Both are made arbitrarily fine by increasing n.

Proposition 6.35 (Two-Simplex Targeting)**.** *A two-block splice traces a line through $(H, \mathrm{CV}_{\mathrm{win}})$ space. To hit both coordinates simultaneously, three blocks suffice: let A, B, C be strips at distinct operating points. The splice $A^aB^bC^c$ with $a+b+c=n$ achieves any point in the triangle* $\mathrm{conv}\{(\mu(A),\mu(B),\mu(C))\}$ *at resolution $O(1/n)$.*

6.13 The $E[c^2]$ Lattice

The per-reel second moment of the count distribution determines the diagonal contribution to $\mathrm{CV}_{\mathrm{win}}$. At $W=3$, a symbol placed as k runs totalling n cells, of which k_1 are singletons, has:

$$\mathbb{E}[c^2 \mid c \geq 1] = \frac{9n - 8k + 2k_1}{n+2k}, \qquad k_1 \in [\max(0, 2k-n),\ \ k-1]. \tag{41}$$

The upper bound $k_1 \le k-1$ is strict: $k_1 = k$ forces $n = k$ (every run is a singleton). The maximum-variance layout at fixed (n, k) is *determined*: $k-1$ singletons plus one run of length $r = n-k+1$, giving:

$$k = \frac{T-n}{W-1}, \qquad r = n-k+1, \tag{42}$$

where T is the total coverage (number of windows containing the symbol). This layout is not searched — it is the unique maximizer of (41) at given (n, T).

Remark 6.36 (The k_1 Residual). At fixed (n, p, k, c) and fixed hit rate, the remaining freedom is k_1 per symbol per reel. This is the *fine* $\mathrm{CV}_{\mathrm{win}}$ lever: closed-form range $\times 1.12$, operating at frozen RTP and frozen hit rate.

Remark 6.37 (The Clustering Profile as a CV Lever). The per-(symbol, reel) clustering vector $\mathbf{c} = (c_f^{(i)})$ is a higher-dimensional freedom than the k_1 residual. At fixed global RTP and fixed global hit rate, different per-symbol c_f vectors can produce the same aggregate H (because H is a function of the per-symbol presences $p_f = m_f/c_f$, and many $\mathbf{c}$ vectors map to the same hit rate via the inclusion–exclusion). Each such vector gives a different $\mathrm{CV}_{\mathrm{win}}$, because the per-reel second moments $m_{2,f,i} = \tau_{f,i}^2 + c_{f,i}^2$ depend on c_f. Treating $\mathbf{c}$ as a scalar (uniform clustering across all fillers) sweeps a one-dimensional line through this space; the achievable $\mathrm{CV}_{\mathrm{win}}$ range along that line is a *lower bound* on the full range. With F fillers on R_{free} non-blocker reels, the profile vector has $F \cdot R_{\mathrm{free}}$ free coordinates. The residual decomposition is: fix counts (RTP), fix the profile $\mathbf{c}$ and arrangement (hit rate), and k_1 per symbol is all that remains. If the profile itself is allowed to vary at fixed globals, the reachable $\mathrm{CV}_{\mathrm{win}}$ range is strictly larger.

7 Hit-Rate Targeting via Co-location

The skeleton (Section 4) fixes the designed symbols' placements. The filler allocation (Section 5) fixes the filler counts at an operating point on the RTP iso-surface. The choice of operating point is the *coarse* hit-rate control: different count allocations place different amounts of filler mass on each reel, changing per-reel m values and hence the achievable presence range. At the chosen operating point, the remaining freedom is the *arrangement* of fillers among the free positions — specifically, the clustering c_f per filler per reel on non-blocker reels (Definition 5.1). Clustering moves $p_f = m_f/c_f$ at fixed m_f, which is the *fine* hit-rate control. The splice construction mixes two arrangements to hit the target exactly.

The co-location matrix Q is both the analysis and construction object: it characterizes the design space, proves RTP-invisibility, and provides the per-symbol clustering targets that the construction realizes. Splice linearity (Proposition 6.32) fills lattice gaps between nearby Q-targeted strips. This section shows that the arrangement freedom is sufficient to realize the target hit rate at the chosen operating point, with RTP preserved exactly.

7.1 Filler Presences: Fixed and Free

Once the filler allocation from Section 5 is committed, each filler f has a known stop count n_f on each reel. On blocker reels (reel $d_f + 1$), fillers are singles ($c_f = 1$), so the presence is fixed by counts:

$$p_f = m_f = \frac{W n_f}{L}. \tag{43}$$

On non-blocker reels $(1, \ldots, d_f)$, the clustering $c_f \ge 1$ is a design parameter (Definition 5.1), giving $p_f = m_f/c_f$. Higher c_f means fewer windows contain the symbol, reducing p_f and hence reducing the total hit rate. This is the primary arrangement-level hit-rate control, operating at frozen m_f (and hence frozen RTP).

7.2 The Off-Diagonal as the Hit-Rate Handle

The game hit rate (Proposition 6.17) is:

$$H = \underbrace{\sum_s \prod_{i=1}^{d} p_s^{(i)}}_{\text{order-1, fixed}} - \underbrace{\sum_{s<t} \prod_{i=1}^{d} \frac{Q_i[s][t]}{L_i}}_{\text{order-2, designable}} + \underbrace{\text{order-3+}}_{\text{small residual}} . \tag{44}$$

The order-1 terms are fixed by counts. The order-2 terms are functions of the off-diagonal entries of Q, which are the design freedom. The order-3+ terms are not controlled by Q but are bounded by the inclusion–exclusion tail (products of per-reel presences with alternating signs).

Rearranging fillers changes which fillers share windows, which changes the off-diagonal entries $Q[f_1][f_2]$ (filler-filler) and $Q[s][f]$ (skeleton-filler). Each such change moves the order-2 hit rate $H^{(2)}$.

7.3 Achievable Hit-Rate Range

Definition 7.1 (Hit-Rate Band)**.** The *hit-rate band* at fixed counts is the interval $[H^{(2)}_{\min}, H^{(2)}_{\max}]$ of order-2 hit rates achievable by varying the filler arrangement. The off-diagonal entries $Q[s][t]$ live in a transportation polytope:

$$\mathcal{P} = \{Q_{st} \geq 0,\ Q_{st} = Q_{ts},\ \textstyle\sum_{t \neq s} Q_{st} = r_s,\ Q_{st} \leq \min(Q_{ss}, Q_{tt})\}, \tag{45}$$

where $r_s = (W-1) \cdot Q[s][s]$ is the margin for symbol s. The hit-rate band is an outer bound on the achievable $H^{(2)}$ values. By Schur-convexity of $\sum x^d$, the minimum of $\sum (Q_{st}/L)^d$ over $\mathcal{P}$ is attained by water-filling (as-equal-as-possible), computable in $O(F^2 \log F)$. The maximum is attained at a vertex of $\mathcal{P}$ (convex maximization over a polytope), computable by vertex enumeration.

Remark 7.2 (Q as the Hit-Rate Specification). The co-location matrix Q is the complete order-2 specification for hit rate. Its diagonal entries $Q[s][s] = Wn_s$ encode the symbol counts; its off-diagonal entries $Q[s][t]$ encode the pairwise arrangement. Both are targeted as part of a single construction. At fixed Q, the order-2 hit rate $H^{(2)}$ is exactly determined; the residual from higher-order terms depends on triple co-occurrence (measured at $\sim 10^{-4}$ on the worked example).

7.4 RTP Preservation

Proposition 7.3 (Hit-Rate Targeting Preserves RTP)**.** *Any rearrangement of fillers that preserves all filler counts preserves RTP exactly.*

Proof. Filler rearrangement changes the off-diagonal block of Q but preserves all counts and all diagonal entries (Proposition 6.10). The RTP formula (7) reads each reel through $m_i = Wn_i/L$ (on paying reels, fixed by counts) and p_i (on blocker reels). For fillers on blocker reels (singles), $p_i = Wn_i/L$, also fixed by counts. For designed symbols, p_i is fixed by the skeleton. No quantity read by the RTP formula changes. Therefore RTP is preserved exactly under rearrangements among fillers. $\square$

Remark 7.4 (Wild Caveat). In the presence of wilds, the effective blocker presence $p_s^{\text{eff}} = p_s + p_{\text{wild}} - Q[s][\text{wild}]/L$ reads the skeleton-filler entry $Q[s][\text{wild}]$. Filler-only rotations (four filler symbols $a, b, c, e \notin \{\text{wild}\}$) change only filler-filler entries and leave every $Q[f][\text{wild}]$ untouched, preserving p_f^{eff} and hence RTP. This requires $F \geq 4$ fillers.

7.5 Order-3 Residual and Directed Correction

The order-2 hit rate $H^{(2)}$ is a function of Q alone (Corollary 6.19). The full hit rate H includes order-3+ terms not captured by Q. After rearranging fillers to achieve a target Q (and hence a target $H^{(2)}$), the actual hit rate H may differ from $H^{(2)}$ by the order-3 residual.

Remark 7.5 (Residual Magnitude and Resolution). The inclusion–exclusion for the total hit rate terminates exactly at order W. At $W = 3$, the order-3 residual between $H^{(2)}$ (from Q) and the true H depends on triple co-occurrence across reels. On the worked example design (Section 15), the measured residual is $|H - H^{(2)}| \approx 10^{-4}$; the magnitude is small because the order-3 terms are products of three per-reel presences with alternating signs and blocker attenuation. This residual is not a fundamental limitation: the splice construction (Proposition 6.32) hits the true H directly, bypassing Q as an intermediary. At n blocks, the HR resolution is $O(1/n)$. At $n = 5$, the error is $\sim$0.02%. At $n = 50$, $\sim$0.002%. Precision costs strip length, linearly. The target is hit exactly when it is a rational multiple of $1/n$. No iteration, no cell swaps, no convergence argument — more blocks is the closed-form answer.

Proposition 7.6 (Exact Hit-Rate Construction)**.** *For any target H^* in the achievable hit-rate band, there exists a finite splice $A^{n_A}B^{n_B}$ with $|H(A^{n_A}B^{n_B}) - H^*| < \epsilon$ for any $\epsilon > 0$, where $n = n_A + n_B = O(1/\epsilon)$. RTP is preserved exactly (all blocks share the same counts). The construction is deterministic: build the two extreme-point strips, compute the ratio, concatenate. Evaluation cost scales as $O(nL_{\text{base}})$ per reel, which is linear in precision — at $n = 50$ ($\epsilon \approx 0.002\%$) with $L_{\text{base}} \approx 2000$, evaluation remains sub-second.*

Remark 7.7 (Two Structural Invariants). Throughout the hit-rate targeting pipeline, two invariants hold:

1. *RTP is preserved exactly*: every operation preserves counts, and RTP reads only counts and the (frozen) skeleton.
2. *The skeleton is untouched*: only filler positions are modified. Every per-symbol target from Section 4 remains exact.

The hit-rate targeting step adds a third achieved target (total hit rate) without disturbing the first two (per-symbol specifications and total RTP).

8 Payout Volatility

8.1 Variance Decomposition

The industry-standard volatility index is defined as the unconditional per-spin standard deviation σ of the payout (scaled by wager), sometimes multiplied by a jurisdiction-specific constant. This metric conflates two distinct player experiences: how often wins occur, and how dramatic they are. The identity

$$\sigma^2 = \frac{\text{RTP}^2}{p}\left(1 - p + \text{CV}_{\text{win}}{}^2\right) \tag{46}$$

decomposes σ^2 into a *zero-inflation* term $(1-p)/p$ driven by hit rate, and a *win-shape* term $\mathrm{CV_{win}}^2$ driven by the conditional payout distribution, where $\mathrm{CV_{win}} = \sigma_{Y|Y>0}/\mathbb{E}[Y|Y>0]$ is the coefficient of variation of the payout given a win.

Proposition 8.1 (Hit-Rate Share of Volatility)**.** *The fraction of σ^2 attributable to hit rate alone is:*

$$f_{HR} = \frac{1-p}{1-p+\mathrm{CV_{win}}^2}. \tag{47}$$

On flat paytables ($\mathrm{CV_{win}} \approx 0.5$*), $f_{HR} \approx 75\%$: hit rate* is *the volatility. On premium paytables (*$\mathrm{CV_{win}} \approx 8$*), $f_{HR} \approx 1.5\%$: the win shape dominates. The paytable determines which regime the game is in.*

Remark 8.2 (Same VI, Different Experience). Three games with identical σ, RTP, and VI can deliver completely different player experiences: frequent small wins with occasional jackpots (high $\mathrm{CV_{win}}$, high p), moderate wins at moderate frequency, or rare uniform payoffs (low $\mathrm{CV_{win}}$, low p). $\mathrm{CV_{win}}$ isolates the win drama; σ does not.

Remark 8.3 (On the Industry Volatility Metric). The volatility index, proportional to σ, is the industry's standard measure of how "swingy" a game feels. But as Proposition 8.1 shows, the majority of σ^2 on typical paytables comes from the hit rate — from the zeros in the payout distribution, not from the wins. A game with a perfectly flat paytable (every win pays identically) can register as "high volatility" simply by having a low hit rate. Yet the player's experience of that game is not swingy: when they win, they always win the same amount. The swings are in the *waiting*, not in the *winning*.

A player does not experience a zero-pay spin as a volatile outcome — it is simply a loss, identical to every other loss. What the player perceives as volatility is the variation *among wins*: sometimes a small return, sometimes a large one. This is precisely what $\mathrm{CV_{win}}$ measures — the dispersion of win sizes relative to the average win.

We propose that $\mathrm{CV_{win}}$ is the natural replacement for the volatility index as a design metric. It is targetable at frozen RTP and hit rate (Section 8.4), pre-computable from the specification (Proposition 8.7), exactly computable from the constructed strip (Proposition 8.5), and it measures the quantity the player actually cares about: how dramatic the wins feel when they hit. The unconditional σ^2 remains derivable from $(\mathrm{RTP}, p, \mathrm{CV_{win}})$ via (46) for any regulatory or analytical purpose that requires it.

Definition 8.4 (Three Design Coordinates)**.** The natural targeting coordinates are $(\mathrm{RTP}, p, \mathrm{CV_{win}})$:

1. RTP is exact from counts (Theorem 2.7).
2. Hit rate p is exact via the inclusion–exclusion formula, which terminates at order W (Section 7).
3. $\mathrm{CV_{win}}$ is the conditional shape parameter, controlled by clustering distribution, filler arrangement, and label assignment (Subsection 8.4).

σ^2 (and hence VI) is determined by their combination via (46). It is a derived quantity, not a design target.

8.2 Exact $\mathrm{CV}_{\mathbf{win}}$ from Design Parameters

Proposition 8.5 (Exact Conditional Volatility)**.** $\mathrm{CV}_{\mathrm{win}}$ *is an exact function of the design parameters, computable in* $O(|S|^2 \cdot R \cdot L)$ *without simulation:*

$$\mathrm{CV}_{\mathrm{win}}{}^2 = \frac{\mathbb{E}[Y^2]}{p \cdot M^2} - 1, \quad M = \frac{\mathrm{RTP}}{p}, \tag{48}$$

where $\mathbb{E}[Y^2]$ *is assembled from per-reel factors computed in a single* $O(L)$ *pass per reel:*

$$\mathbb{E}[Y^2] = \underbrace{\sum_{s,k} v_{s,k}^2 \prod_{i=1}^{k} \hat{m}_{2,s,i} \cdot B_{s,k+1}}_{\textit{diagonal}} + \underbrace{2 \sum_{(s,k)<(t,l)} v_{s,k}\, v_{t,l} \prod_{i} J_i(s,t,k,l)}_{\textit{cross terms}}, \tag{49}$$

where $\hat{m}_{2,s,i} = \frac{1}{L_i} \sum_{\mathrm{pos}} C_{s,i}(\mathrm{pos})^2$ *is the unconditional second moment of the count (summed over all positions in one pass),* $B_{s,k+1}$ *is the blocker factor, and* J_i *is the per-reel joint factor:*

- *Paying reel (both present):* $J_i = \frac{1}{L_i} \sum_{\mathrm{pos}} C_{s,i}(\mathrm{pos}) \cdot C_{t,i}(\mathrm{pos})$.
- *Blocker reel (s absent, t present):* $J_i = \frac{1}{L_i} \sum_{\mathrm{pos}:\, C_s=0} C_{t,i}(\mathrm{pos})$.
- *Both blocked (*$k = l$*, reel* $k{+}1$*):* $J_i = P(C_{s,i} = 0,\ C_{t,i} = 0)$.
- *One ladder ended, other paying (*$k < l$*, reels* $k{+}2 \ldots l$*): s unconstrained, t paying,* $J_i = \frac{1}{L_i} \sum_{\mathrm{pos}} C_{t,i}(\mathrm{pos})$ *(the unconditional first moment of t's count, since only t's ways contribute to the cross term).*
- *One ladder ended, other blocked (*$k < l$*, reel* $l{+}1$*): t blocked,* $J_i = 1 - p_{t,i}$.
- *Both ladders ended (reels past* $\max(k,l){+}1$*):* $J_i = 1$.

Proof. $\mathbb{E}[Y^2] = \mathbb{E}\big[(\sum_{s,k} Y_{s,k} \mathrm{Pay}_{s,k})^2\big]$. Expanding the square and using reel independence, each term factors into per-reel contributions. Each per-reel factor is computed from a single pass over L positions. The same-symbol cross-depth terms vanish (mutually exclusive: a symbol achieves exactly one OAK depth per spin). All per-reel factors are exact expectations over L positions; no approximation is introduced. □

Remark 8.6 (Skeleton–Filler Decomposition). $\mathbb{E}[Y^2]$ decomposes as $\mathbb{E}[Y_{\mathrm{skel}}^2] + 2\,\mathbb{E}[Y_{\mathrm{skel}} \cdot Y_{\mathrm{fill}}] + \mathbb{E}[Y_{\mathrm{fill}}^2]$. The first two terms are fixed by the skeleton. The third is the design freedom controlled by the filler arrangement.

8.3 Pre-Computable $\mathrm{CV}_{\mathbf{win}}$ Bounds

Proposition 8.7 (Theoretical $\mathrm{CV}_{\mathrm{win}}$ Range)**.** *At fixed* $(\mathrm{RTP}, p, \textit{paytable})$*, the achievable* $\mathrm{CV}_{\mathrm{win}}$ *is bounded by the moment problem on the pay-value support. With* $M = \mathrm{RTP}/p$ *and single-event payouts in* $[v_{\min}, v_{\max}]$ *(where* $v_{\max}$ *is the maximum achievable payout including ways, i.e.* $v_{\max} = \max_{s,k} v_{s,k} \cdot W^k$*):*

$$\mathrm{CV}_{\mathrm{win,\,max}}{}^2 = \frac{(M - v_{\min})(v_{\max} - M)}{M^2}. \tag{50}$$

This outer bound applies to the single-event payout distribution and requires no strip. Cross-event contributions (multiple symbols winning simultaneously) are captured by the cross-term sum in (49). The inner bound is obtained by sweeping the three controls below through their attainable ranges and evaluating (49).

8.4 Three Controls for $\mathrm{CV}_{\mathrm{win}}$

All three operate at frozen RTP.

Proposition 8.8 (Control 1: Clustering Distribution (per reel))**.** *At fixed (p_s, c_s) for designed symbol s on reel i, the gap pattern within the cluster determines $\tau^2_{s,i}$. The per-reel second moment $m_2 = \tau^2 + c^2$ ranges over a computable interval. At $W = 3$ with count levels $\{1, 2, 3\}$, the fraction α_3 of count-3 windows ranges from $\max(0, c - 2)$ to $(c - 1)/2$, giving:*

$$m_2 \in \bigl[3c - 2 + 2\max(0, c{-}2),\ \ 4c - 3\bigr]. \tag{51}$$

The compound effect across d paying reels is $\prod_i m_{2,i}$: multiplicative, not additive. RTP and hit rate are unchanged ($m = p \cdot c$ and p are fixed).

Proposition 8.9 (Control 2: Directed Filler Arrangement (per reel))**.** *The total skeleton–filler co-occurrence $\sum_f Q[\text{designed}][f]$ per reel is fixed by the skeleton geometry. The* distribution *across fillers is free: placing expensive fillers (M1, M2) in windows adjacent to the premium designed symbol (H1) increases the cross terms in (49); placing cheap fillers (9, 10) there decreases them. Intuitively, co-locating high-paying symbols creates windows where multiple large pays fire simultaneously, amplifying win-shape dispersion; spreading them apart makes large co-occurrences rarer and reduces volatility. This compounds across paying reels through the per-reel joint factor J_i. RTP is unchanged (counts are invariant); hit rate is preserved by restricting to filler-only rearrangements (Remark 7.4).*

Proposition 8.10 (Control 3: Label Permutation (global))**.** *Among fillers with identical per-reel count profiles, permuting pay-value assignments preserves RTP, Q, and hit rate, changing only the payout variance. The change operates through third-party cross terms: for each other symbol g, the contribution shifts by $v_g(v_{f_2} - v_{f_1})(Q[f_1][g] - Q[f_2][g])$. This is a coarse, one-time, global control.*

Remark 8.11 (Combined Leverage). All three controls compound. Controls 1 and 2 are per-reel and multiply across paying reels. Control 3 is global. The combined $\mathrm{CV}_{\mathrm{win}}$ range at fixed RTP, HR, and skeleton is determined by the product of per-reel m_2 ranges (Proposition 8.8), the filler arrangement freedom within the skeleton co-occurrence budget (Proposition 8.9), and the label permutation set (Proposition 8.10). The range scales with the paytable's pay dispersion: higher $v_{\max}/v_{\min}$ ratios amplify all three controls.

8.5 Variance from Count Allocation

Proposition 8.12 (Count Allocation as the Primary σ^2 Control)**.** *From the decomposition (46), σ^2 depends on p through $\mathrm{RTP}^2(1 - p)/p$, which varies by orders of magnitude across the achievable hit-rate range. The filler count allocation (Section 5) moves p on the RTP iso-surface; this is the dominant σ^2 control. The three $\mathrm{CV}_{\mathrm{win}}$ controls provide the* shape *adjustment at each chosen hit rate.*

Proposition 8.13 (Pre-Computable Variance Interval)**.** *The achievable σ^2 interval at fixed RTP is computable from the paytable and filler budget:*

1. *The filler allocation set is a product of R box-constrained simplices (Corollary 5.4). Along any edge, the RTP is affine (Lemma 5.2).*

2. *Find crossing points where $\mathrm{RTP}(t) = R^*$.*

3. *At each crossing point, evaluate* p *and the* $\mathrm{CV}_{\mathrm{win}}$ *range from Proposition 8.7.*

4. *Compute* σ^2 *from (46) at the extremes of both* p *and* $\mathrm{CV}_{\mathrm{win}}$*.*

This gives an inner approximation. The number of edge evaluations is $R \cdot \binom{F}{2} \cdot V^{R-1}$*, independent of* L*, where* V *is the number of vertices of each box-constrained simplex factor.*

8.6 Concentration Bounds

Proposition 8.14 (Concentration of the Empirical Distribution)**.** *With* σ^2 *exactly known and the per-spin payout bounded by* b*, Bernstein's inequality [14] gives:*

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^{n} Y_i - \mathrm{RTP}\right| \geq t\right) \leq 2\exp\left(-\frac{nt^2}{2\sigma^2 + \frac{2}{3}bt}\right). \tag{52}$$

8.7 Invariant Summary

At the conclusion of targeting, four quantities are simultaneously achieved:

1. *Per-symbol specifications* (p, c, τ^2): fixed by the skeleton (Section 4).

2. *Total RTP*: fixed by count allocation (Section 5), preserved by all subsequent operations.

3. *Total hit rate*: realized by co-location construction (Section 7), preserved by hit-rate-invariant operations.

4. *Conditional payout shape* $\mathrm{CV}_{\mathrm{win}}$: targeted by clustering distribution, directed filler arrangement, and label permutation, all at frozen RTP and hit rate.

σ^2 and VI are derived from $(\mathrm{RTP}, p, \mathrm{CV}_{\mathrm{win}})$ via (46).

9 Full Construction Theorem

This section assembles the results of Sections 4–8 into a single existence and constructibility theorem for complete games.

9.1 The Design Parameter Space

Definition 9.1 (Design Parameter Space)**.** The *design parameter space* $\mathcal{D}$ is the set of all tuples $(\mathbf{n}, \boldsymbol{\tau}^2, Q_1, \ldots, Q_R)$ where:

1. $\mathbf{n} = (n_s^{(i)})$ is a matrix of per-symbol, per-reel stop counts satisfying $\sum_s n_s^{(i)} = L_i$ for each reel i.

2. $\boldsymbol{\tau}^2 = (\tau_s^{(i)})$ is a matrix of per-symbol, per-reel conditional count variances. The attainable τ^2 values at fixed (p, c) form a finite, computable set determined by the gap-pattern compositions of the cluster. The moment-problem bounds $[\tau^2_{\min}, \tau^2_{\max}]$ (Proposition 3.8) provide an outer envelope; the realizable values are a subset.

3. Q_i is a co-location matrix for reel i, consistent with the counts $\mathbf{n}^{(i)}$ and the skeleton structure, achievable by a physical strip (verified constructively).

9.2 The Target Map

Definition 9.2 (Target Map)**.** The *target map* $\Phi : \mathcal{D} \to \mathbb{R}^3$ sends each design parameter tuple to:

$$\Phi(\mathbf{n}, \boldsymbol{\tau}^2, Q_1, \ldots, Q_R) = (\text{RTP},\ H^{(2)},\ \text{CV}_{\text{win}}), \tag{53}$$

where RTP is computed from $\mathbf{n}$ alone (Proposition 2.11), $H^{(2)}$ is the order-2 hit rate computed from $\{Q_i\}$ (Corollary 6.19), and CV_{win} is computed from $\mathbf{n}$, $\boldsymbol{\tau}^2$, and $\{Q_i\}$ (Proposition 8.5). RTP and CV_{win} are exact functions of the design parameters. $H^{(2)}$ is an exact function of Q; the true hit rate H includes higher-order terms not captured by Q, with the residual $H - H^{(2)}$ measured at $\sim 10^{-4}$ on the worked example (the magnitude depends on the design's per-reel presences and pay depths). The splice construction (Proposition 6.32) targets the true H directly at resolution $O(1/n)$.

Definition 9.3 (Achievable Set)**.** The *achievable set* $\mathcal{A}$ is $\Phi(\mathcal{D}) \subseteq \mathbb{R}^3$, parameterized by the skeleton and the resulting filler budget $N_i = L_i - \sum_{s \in \Omega_{\text{designed}}} n_s^{(i)}$ on each reel. A target specification, consisting of per-symbol targets $(p_s^{(i)}, c_s^{(i)}, \tau_s^{2(i)})$ satisfying the packing constraint (54) and global targets $(R^*, H^*, \text{CV}_{\text{win}}^*)$, is *feasible* if the per-symbol targets lie within their attainable sets (Proposition 3.5 for (p, c); gap-pattern enumeration for τ^2) and $(R^*, H^*, \text{CV}_{\text{win}}^*) \in \mathcal{A}$.

9.3 The Full Construction Theorem

Theorem 9.4 (Full Construction)**.** *Let each designed symbol s be assigned per-reel targets $(p_s^{(i)}, c_s^{(i)}, \tau_s^{2(i)})$ within their attainable ranges (Sections 3–4), subject to the packing constraint that the total designed-symbol footprint leaves positive filler budget on each reel:*

$$\sum_{s \in \Omega_{\text{designed}}} \text{sp}_s^{(i)} < 1 \quad \textit{for all reels } i, \tag{54}$$

where $\text{sp}_s^{(i)} = p_s^{(i)} c_s^{(i)} / W$ is the symbol percent of s on reel i (Proposition 4.7). Let $(R^*, H^*, \text{CV}_{\text{win}}^*)$ be global targets. The construction pipeline proceeds as follows:

1. RTP band *(pre-computable): R^* must lie in $[\text{RTP}_{\min}, \text{RTP}_{\max}]$ (Definition 5.5). If not, the target is infeasible at these skeleton counts; the binding constraint is reported.*

2. HR outer bound *(pre-computable): H^* must lie in the hit-rate band at the filler counts determined by R^* (Definition 7.1). This bound is computed from the transportation polytope of pairwise co-locations. At finite L, integrality and connectivity may exclude some interior points; concatenation K times (Proposition 4.11) multiplies all Q entries by K, making the integer lattice K-fold finer without changing any game metric. At sufficient K, the integer-achievable set fills the continuous bound to $O(1/KL)$ resolution. Targets outside the band are certified infeasible at any K.*

3. CV_{win} targeting*: CV_{win}^* is achieved at the chosen (R^*, H^*) via clustering distribution, directed filler arrangement, co-location along the HR-preserving kernel, and multiplier value distribution (Section 8.4). All controls operate on integer lattices with resolution $O(1/L)$; at strip length $L = O(1/\varepsilon)$, any target in the achievable range is hit to precision ε. Splicing provides additional $O(1/n)$ refinement between lattice points.*

4. Constructive certificate*: extreme-point strips are built at the HR band endpoints, and the target $(H^*, \text{CV}_{\text{win}}^*)$ is reached by splicing at the appropriate ratio (Proposition 6.32). Precision is $O(1/n)$ where n is the number of blocks.*

RTP is achieved exactly. Hit rate and $\mathrm{CV}_{\mathrm{win}}$ *are achieved to* $O(1/L)$ *precision from the integer lattice of co-location entries, clustering values, and multiplier assignments. At* $L = O(1/\varepsilon)$*, all three coordinates are within* ε *of their targets.* σ^2 *is derived from* $(\mathrm{RTP}, p, \mathrm{CV}_{\mathrm{win}})$ *via (46).*

The pipeline is deterministic: it either produces a strip or reports which bound is violated. Constructibility of any candidate Q *is decidable before a strip is built (*$\sim$60 ms *to reject,* $\sim$2.6 s *to accept and build) (Proposition 6.27, Section 6.11). Any target between two constructively verified operating points is achievable at* $L = O(1/\varepsilon)$ *for precision* ε*, by lattice density and splice linearity. The full pipeline from specification to verified strips runs in* $\sim$17 s.

Proof. The proof follows the pipeline steps, showing that each preserves the targets established by previous steps.

Step 1: Skeleton. For each designed symbol s on each reel i, the target $(p_s^{(i)}, c_s^{(i)}, \tau_s^{2(i)})$ determines a quasi-contiguous cluster with a specific span (fixing p and c) and a specific gap ordering (fixing τ^2). By Theorem 4.9, this placement exists and is constructible. Each symbol's statistics depend only on its own stops (Proposition 4.4), so clusters may be separated or interleaved as the design requires.

Step 2: Filler counts. Since R^* lies in the RTP band, the intermediate value theorem on the convex allocation polytope (Proposition 5.8) guarantees a filler count vector $\mathbf{n}^*$ with $\mathrm{RTP}(\mathbf{n}^*) = R^*$ at lattice resolution $O(1/L)$.

Reachability. By Corollary 6.33, the achievable set of $(H, \mathbb{E}[Y^2])$ pairs at fixed counts is reachable by mixing (Corollary 6.33). Any target in the interior is reached by splicing extreme-point strips (Proposition 6.32). The derived coordinate $\mathrm{CV}_{\mathrm{win}}$ is a smooth function of these, so any interior $(H, \mathrm{CV}_{\mathrm{win}})$ target is reachable. Resolution is $O(1/n)$ where n is the number of splice blocks, made arbitrarily fine by increasing n.

Step 3: Filler arrangement. The target hit rate H^* is achieved in two stages:

(a) Direct construction. For each filler f on each non-blocker reel, the clustering c_f is chosen so that $p_f = m_f/c_f$ yields the target per-reel presences. Since $m_f = Wn_f/L$ is fixed by counts, the designer solves for the c_f values that produce H^* via the inclusion–exclusion formula. The filler stops are then placed as quasi-contiguous clusters at the target c_f, exactly as for designed symbols. This produces a single strip achieving the target HR at any lattice point where the required c_f values are realizable (integer-compatible gap patterns).

(b) Continuous interpolation via splicing. When H^* falls between two lattice points, two strips A and B at nearby achievable HR values are spliced: $A^{n_A}B^{n_B}$ achieves any rational convex combination of $H(A)$ and $H(B)$ at resolution $O(1/n)$ (Proposition 6.32). Both strips share the same skeleton and counts, preserving RTP exactly. The splice fills the gaps in the lattice, making the achievable set dense rather than discrete.

In both cases, RTP is preserved exactly (counts are invariant) and Proposition 6.32 guarantees the splice mixture is exact when the skeleton pins the boundary cells.

Step 4: Verification. The constructed strips realize:

- $\mathrm{RTP} = R^*$: determined by counts alone, fixed in Step 2, preserved by Step 3.
- $H = H^*$: the splice ratio targets the true hit rate directly at resolution $O(1/n)$ (Proposition 7.6). Precision costs strip length, linearly.
- $\mathrm{CV}_{\mathrm{win}} = \mathrm{CV}^*_{\mathrm{win}} + O(1/L)$: all CV controls (clustering, co-location, multiplier distribution, k_1) operate on integer lattices with $O(1/L)$ step size. Any target between two constructively verified CV values is achievable at $L = O(1/\varepsilon)$.

All four steps are deterministic and finite. □

9.4 Pre-Computability of the Achievable Set

Proposition 9.5 (The Achievable Set Is Pre-Computable)**.** *The achievable set $\mathcal{A}$ can be computed from the paytable, the skeleton specification, and the reel lengths, before any strip is constructed:*

1. *The* RTP range *is the interval* $[\mathrm{RTP}_{\min}, \mathrm{RTP}_{\max}]$ *from the filler allocation polytope (Definition 5.5).*
2. *At each RTP value, the* hit-rate range *is determined by the co-location freedom at those counts (Definition 7.1).*
3. *At each (RTP, hit-rate) pair, the* volatility range *is determined by the count allocation on the RTP iso-surface (Section 8.5), the attainable τ^2 range (Proposition 3.8), and the hit-rate-preserving rotation freedom (Theorem 6.20).*

Under the continuous relaxation $\tilde{\mathcal{D}}$ (real-valued counts on the allocation polytope, τ^2 on the moment interval, Q on the LP relaxation of $\mathcal{F}$), the image $\tilde{\mathcal{A}} = \Phi(\tilde{\mathcal{D}})$ is compact and connected. The integer-achievable subset $\mathcal{A}_{\mathbb{Z}}$ is an $O(1/L)$ net in $\tilde{\mathcal{A}}$ (Proposition 9.6). The LP relaxation is a necessary condition for the Q coordinate ; the complete test requires integrality and connectivity .

9.5 Integrality and Resolution

Proposition 9.6 (Lattice Resolution)**.** *All design parameters are ultimately integers (stop counts, Q entries, gap positions). The continuous achievable set $\mathcal{A}$ is approximated by the integer-achievable subset $\mathcal{A}_{\mathbb{Z}}$ to resolution $O(1/L)$ in each coordinate. As $L \to \infty$, $\mathcal{A}_{\mathbb{Z}}$ becomes dense in $\mathcal{A}$, and every target in the interior of $\mathcal{A}$ is achievable exactly.*

Proof. The RTP resolution is $O(W/L)$ per count change (Proposition 5.8). The τ^2 resolution is $O(1/a)$ per gap reordering, where $a = O(L)$. The Q entry resolution is integer-valued with $Q[s][t] \in \{0, 1, \ldots, L\}$, giving per-entry resolution $O(1/L)$ in the normalized quantity $Q[s][t]/L$. Since L is a derived quantity with no upper bound (Proposition 4.8), the resolution can be made arbitrarily fine. □

Proposition 9.7 (Joint HR/Volatility Targeting at Fixed RTP)**.** *Given a strip achieving target RTP and $H^{(2)}$, the conditional payout shape* $\mathrm{CV}_{\mathrm{win}}$ *(and hence σ^2) can be adjusted to arbitrary precision without changing either RTP or $H^{(2)}$, by the following procedure:*

1. Concatenate *the strip K times (Proposition 4.11). This preserves* RTP, $H^{(2)}$, *and σ^2 exactly (all normalized quantities $Q[s][t]/L$ are unchanged), while multiplying the number of integer Q values by K.*
2. Apply hit-rate-preserving rotations *satisfying condition (37). At pay depth $d = 3$, solutions to condition (37) are abundant at every concatenation factor K (Remark 9.8). Each rotation is realized by a singles-preserving transposition on the strip (no two stops of the same filler land within $W - 1$ positions after the swap). Each such rotation changes σ^2 by $O(1/L)$, and $L = KL_0$ is now K times larger, providing K-fold finer variance resolution. For games with mixed pay depths, the available rotations are depth-specific ($d = 3$-preserving but not $d = 2$-preserving); a single non-consecutive $\delta = 1$ seed move breaks the mod-K lattice on the off-diagonal entries, restoring the availability of consecutive-entry (depth-universal) rotations for all subsequent moves. The seed move introduces a one-time $O(1/L)$ perturbation to $H^{(2)}$ at non-target depths, correctable by a subsequent compensating move.*

Since RTP depends only on counts (Theorem 2.7) and counts are preserved by both concatenation and rotations, RTP is exact throughout. Since $H^{(2)}$ depends on $\sum(Q[s][t]/L)^d$ and rotations satisfying condition (37) preserve this sum at the relevant pay depth, $H^{(2)}$ is preserved throughout. The variance resolution is $O(1/KL_0)$, arbitrarily fine for sufficiently large K.

Remark 9.8 (Physical Operations vs. Algebraic Scaling). A natural objection is that concatenation scales every Q entry by K, so any rotation must use $\delta = K$ to preserve divisibility, yielding a variance step of $K/(KL_0) = 1/L_0$ — no improvement. This reasoning is incorrect. Concatenation is a strip-level operation; rotations are physical operations on the strip. A single transposition of two filler stops in one copy of the concatenated strip changes exactly those windows containing the swapped positions, modifying the relevant Q entries by ± 1 regardless of L. The variance step is $(v_a - v_c)(v_b - v_e)/L = (v_a - v_c)(v_b - v_e)/(KL_0)$, which genuinely shrinks with K. After such a move, the Q entries are no longer multiples of K (e.g., $5K \to 5K \pm 1$), and condition (37) admits $\delta = 1$ solutions on non-K-divisible entries. Existence: at $K = 2$, the quadruple $(A, C, E, B) = (0, 6, 4, 6) \to (1, 7, 3, 5)$ satisfies $(A^3 + C^3) - (E^3 + B^3) = 0$. After such a move, entries are no longer K-divisible, and consecutive-entry solutions become available.

9.6 Infeasibility Detection

Remark 9.9 (When Targets Are Infeasible). If the target specification lies outside the feasible set, the construction pipeline detects this at the step where the remaining freedom cannot accommodate the remaining targets:

- Per-symbol footprints violate the packing constraint (54): detected before construction begins.
- R^* outside the RTP band at the resulting filler budget: detected in Step 2.
- H^* outside the hit-rate band at the required counts: detected in Step 3.
- $\mathrm{CV}^*_{\mathrm{win}}$ outside the achievable $\mathrm{CV}_{\mathrm{win}}$ range at the required (counts, Q): detected in Step 1 (clustering) or Step 3 (rotations).

In each case, the pipeline reports which constraint binds and the achievable range at that constraint, enabling the designer to adjust the infeasible target.

10 Wilds

A *wild symbol* substitutes for every paying symbol. This section shows that the wild enters the framework through two additive corrections: one to each paying symbol's mean count m, and one to the effective co-location matrix Q. Both are closed-form. The moat around the wild cluster handles the RTP interaction.

10.1 Linear Separability of Mean Count

Proposition 10.1 (Effective Mean Count). *For a paying symbol s, the effective mean count per window is:*

$$m_s^{\mathrm{eff}} = m_s^{\mathrm{native}} + m_{\mathrm{wild}}, \tag{55}$$

where $m_s^{\mathrm{native}} = Wn_s/L$ and $m_{\mathrm{wild}} = Wn_{\mathrm{wild}}/L$. The wild's contribution is additive and arrangement-independent.

Proof. By linearity of expectation: $\mathbb{E}[C_s^{\mathrm{eff}}] = \mathbb{E}[C_s^{\mathrm{native}}] + \mathbb{E}[C_{\mathrm{wild}}] = m_s^{\mathrm{native}} + m_{\mathrm{wild}}$. □

10.2 Additive Correction to the Co-location Matrix

Proposition 10.2 (Effective Co-location). *Every window containing the wild effectively contains every paying symbol (via substitution). The effective co-location matrix for paying symbol pairs is:*

$$Q_{\text{eff}}[s][t] = Q_{\text{native}}[s][t] + Q[\text{wild}][\text{wild}] - Q_{\text{overlap}}[s][t] \quad \textit{for all paying pairs } s \neq t, \tag{56}$$

where $Q[\text{wild}][\text{wild}]$ is the number of windows containing the wild (fixed by the skeleton), and $Q_{\text{overlap}}[s][t]$ counts windows containing the wild AND both s and t natively. The overlap term prevents double-counting of windows at the edges of the wild cluster where native fillers and the wild coexist. The overlap count is pre-computable from the skeleton geometry and is $O(1)$ per wild cluster (at most $2(W-1)$ windows per cluster, corresponding to the entry and exit edges).

Proof. A window either contains the wild or it does not. In windows without the wild, only native co-occurrences contribute. In windows with the wild, every paying symbol is effectively present, so every pair co-occurs. Summing $Q_{\text{native}}[s][t] + Q[\text{wild}][\text{wild}]$ double-counts windows where both s and t are natively present AND the wild is present. Subtracting $Q_{\text{overlap}}[s][t]$ corrects this. □

Remark 10.3. For highly clustered wilds ($c_{\text{wild}} \geq 2$), most wild-containing windows have ≥ 2 wild stops and at most one native symbol, so $Q_{\text{overlap}}[s][t] = 0$ for most pairs. The correction is nonzero only for filler pairs that appear together in the $2(W-1)$ edge windows per cluster. The correction is $O(K/L)$ per pair (where K is the number of wild clusters), exactly computable, and must be included for correctness.

Corollary 10.4 (Effective Diagonal). *The effective diagonal (per-symbol presence count) is:*

$$Q_{\text{eff}}[s][s] = Q_{\text{native}}[s][s] + Q[\text{wild}][\text{wild}] - Q[s][\text{wild}], \tag{57}$$

where $Q[s][\text{wild}]$ counts windows containing both native s and wild (by inclusion-exclusion). Under the moat ($Q[s][\text{wild}] = 0$ for non-adjacent symbols), this simplifies to $Q_{\text{native}}[s][s] + Q[\text{wild}][\text{wild}]$.

10.3 RTP and the Moat

Corollary 10.5 (RTP Under Wild Presence). *RTP is computed by substituting m_s^{eff} for m_s in the RTP formula (7), with one correction: all-wild ways (no native symbols on any paying reel) are counted once per paying symbol but awarded once at the highest pay s^*. The correction subtracts the overcounted all-wild contributions:*

$$\Delta\text{RTP} = -\sum_{s \neq s^*} \sum_{k} v_k^{(s)} \cdot \prod_{i=1}^{k} m_{\text{wild}}^{(i)} \cdot (1 - p_s^{\text{eff},(k+1)}). \tag{58}$$

When the wild has no presence on reel 1, $\prod m_{\text{wild}}^{(i)} = 0$ and the correction vanishes. The effective blocker presence $p_s^{\text{eff}} = p_s + p_{\text{wild}} - Q[s][\text{wild}]/L$ is determined by the skeleton geometry. Under the moat, fillers adjacent to the wild cluster have $Q[f][\text{wild}] > 0$ (they share edge windows with the wild), producing the degenerate window correction of Proposition 10.6. This is pre-computable from the skeleton before any filler is placed.

10.4 Degenerate Window Correction

Proposition 10.6 (Degenerate Windows from Wild Clustering). *When the wild has $c_{\text{wild}} > 1$, some windows contain multiple wild stops and fewer than W distinct symbols. For a filler symbol s adjacent to the wild cluster (in the moat), let d_s denote the co-occurrence deficit: the count of degenerate windows containing s, weighted by the shortfall in distinct neighbors. The margin law becomes:*

$$\sum_{t \neq s} Q[s][t] = (W-1) \cdot Q[s][s] - d_s. \tag{59}$$

The deficit d_s is an integer computable from the wild's cluster structure. At $W = 3$, each degenerate window contributes deficit 1.

10.5 Wild as a Skeleton Symbol

The wild is placed in the skeleton with its own per-reel targets $(p^{(i)}_{\text{wild}}, c^{(i)}_{\text{wild}}, \tau^{2(i)}_{\text{wild}})$, subject to the same constructibility results as any designed symbol (Theorem 4.9). Its footprint enters the packing constraint (54). The moat (Definition 4.3) separates the wild cluster from other skeleton symbols, ensuring $Q[\text{wild}][s] = 0$ for other designed symbols. Fillers fill the remaining positions, including moat positions adjacent to the wild.

10.6 Framework Compatibility

Proposition 10.7 (Framework Under Wilds). *The framework of Sections 2–9 extends to games with wild symbols via two additive corrections:*

1. *$m^{\text{eff}}_s = m_s + m_{\text{wild}}$ replaces m_s in the RTP formula, with the all-wild overcounting correction (Corollary 10.5).*
2. *$Q_{\text{eff}}[s][t] = Q_{\text{native}}[s][t] + Q[\text{wild}][\text{wild}] - Q_{\text{overlap}}[s][t]$ replaces $Q[s][t]$ in the hit-rate formula (Proposition 10.2), with the degenerate window correction for moat fillers (Proposition 10.6).*

Both corrections are pre-computable from the skeleton. The filler-filler block of Q_{native} retains its full design freedom. The Full Construction Theorem (Theorem 9.4) holds with the wild included in Ω_{designed}.

Remark 10.8. Kamanas et al. [5] noted that their algorithm's behavior "in the case where special symbols appear in reels is also unknown." In the present framework, wilds are two additive corrections to m and Q, both closed-form, both pre-computable from the skeleton geometry.

11 Value-Bearing Symbols

A *value-bearing symbol* is a symbol whose stops carry a numerical value in addition to their identity. Common instances include cash amounts (scatter pays) and multipliers (wild multipliers), but the framework applies to any symbol whose stops carry a designable numerical parameter. The symbol's identity determines where it participates in OAK evaluation. The numerical value determines how much it contributes when it does. This section shows that the numerical values are independently targetable via a linear system over the stop values, and that the attainable set of per-count averages is pre-computable.

11.1 Definition and Evaluation

Definition 11.1 (Value-Bearing Symbol)**.** A *value-bearing symbol* s has, on each reel i, a set of $n_s^{(i)}$ stops, each carrying a numerical value $\omega_j \in \mathbb{R}_{\geq 0}$. The symbol's identity (s) determines its coverage p, conditional count c, and count distribution τ^2 via the framework of Sections 2–4. The numerical values $\{\omega_j\}$ determine the expected payout conditional on the symbol's count in a window.

Definition 11.2 (Per-Count Average)**.** For a value-bearing symbol with count k in a window (i.e., k of the W cells contain the symbol), the *per-count average* $\bar{\omega}_k$ is the expected sum (or product, depending on the evaluation rule) of the numerical values of the k visible stops. For additive evaluation (cash scatters):

$$\bar{\omega}_k = \mathbb{E}\left[\sum_{j=1}^{k} \omega_{\text{visible},j} \,\middle|\, \text{count} = k\right]. \tag{60}$$

For multiplicative evaluation (wild multipliers):

$$\bar{\omega}_k = \mathbb{E}\left[\prod_{j=1}^{k} \omega_{\text{visible},j} \,\middle|\, \text{count} = k\right]. \tag{61}$$

11.2 Linear System for Per-Count Averages

The stops of a value-bearing symbol in a quasi-contiguous cluster occupy specific positions. At $W = 3$, windows containing the symbol have counts 1, 2, or 3. Each stop participates in windows of potentially multiple count levels (a stop near the cluster edge appears in count-1 and count-2 windows simultaneously).

Proposition 11.3 (Per-Count Targeting via Linear System)**.** *The per-count averages $(\bar{\omega}_1, \ldots, \bar{\omega}_K)$, where $K \leq W$ is the maximum count in any window, are linear functions of the stop values $(\omega_1, \ldots, \omega_n)$. The coefficient matrix $A \in \mathbb{R}^{K \times n}$ has entry $A_{k,j}$ equal to the fraction of count-k windows containing stop j. When A has full row rank, the system $A\boldsymbol{\omega} = \bar{\boldsymbol{\omega}}$ is solvable for any target in the* attainable polytope $\{A\boldsymbol{\omega} : \boldsymbol{\omega} \geq 0\}$*, with $n - K$ free parameters available for higher-moment targeting or satisfying additional constraints. The attainable polytope is pre-computable from the cluster geometry (Proposition 11.4). For multiplicative evaluation (wild multipliers), substituting $u_j = \log \omega_j$ and targeting the per-count geometric mean $\log \bar{\omega}_k^{\text{GM}} = \sum_j A_{k,j} u_j$ reduces to the same linear system with the same rank conditions. The geometric mean is the natural target for multiplicative evaluation; if the arithmetic mean of products is required, the system is polynomial of degree k in the stop values.*

Proof. The per-count average $\bar{\omega}_k$ is, by definition, the mean over all count-k windows of the sum of visible stop values. This is linear in the stop values: $\bar{\omega}_k = \sum_j A_{k,j}\omega_j$. The coefficient matrix A is determined by the cluster geometry, which is fixed before numerical values are assigned.

Full row rank holds when the count levels are sufficiently differentiated by the cluster geometry. For quasi-contiguous clusters with mixed gaps, A has full row rank at $n \geq W$. For contiguous clusters, full rank requires $n \geq 2W - 1$. For singles ($c = 1$), the system is trivially solvable (one count level, $K = 1$). Rank deficiency occurs only for very small clusters or perfectly symmetric repeating patterns. □

11.3 Attainable Set and Resolution

Proposition 11.4 (Attainable Per-Count Averages). *The attainable set of per-count averages* $(\bar{\omega}_1, \bar{\omega}_2, \bar{\omega}_3)$ *is a polytope determined by the cluster geometry and the bounds on individual stop values* ($\omega_j \geq 0$*, and any upper bound imposed by the game design). The polytope is pre-computable from the skeleton specification.*

Proposition 11.5 (Resolution via Reel Length). *Each stop carries one numerical value, so the granularity of the per-count average is* $O(1/n_k)$ *where* n_k *is the number of stops participating in count-*k *windows. Since* n_k *scales with* L *(longer reels have more stops), the resolution of the per-count averages is* $O(1/L)$*. Doubling the reel length (Proposition 4.11) doubles the number of distinct values available and halves the targeting resolution.*

Remark 11.6 (Weight Tables Are Subsumed). Industry practice assigns values to symbol stops via a *weight table*: a single probability distribution from which every stop's value is drawn independently. When k stops are visible (count $= k$), the expected sum is $k \cdot \mathbb{E}[w]$ and the expected product is $\mathbb{E}[w]^k$. The per-count averages are locked in a ratio determined by k; they cannot be targeted independently.

The linear system of Proposition 11.3 is strictly more powerful. Each physical stop receives its own fixed value, and different stops contribute with different weights to different count levels. The per-count averages are independently targetable because the coefficient matrix generically has full row rank: the count levels see different weighted combinations of stop values. A weight table is the special case where all stops are assigned values from the same distribution; the framework does not require this restriction.

11.4 Distributional Control

Proposition 11.7 (Distributional Control). *At a single count level* k*, the* n_k *stops participating in count-*k *windows contribute to the empirical distribution of visible values with weights* $A_{k,j}$*. Within one count level, targeting* J *moment constraints requires* J *linearly independent columns of row* k *of* A*, which is generically available for* $J < n_k$*. Across count levels simultaneously, the same stop values* ω_j *appear in all rows of* A*, so moment targets at different count levels are coupled through the shared unknowns. The number of simultaneously targetable constraints (moments across all levels combined) is bounded by the rank of the stacked system, which equals* K *(the number of count levels) when* A *has full row rank, and grows with* n *as additional moment equations are appended.*

Proof. Each moment target at count level k adds one row to the constraint matrix. The j-th moment at level k is $\sum_i A_{k,i}^{(j)} \omega_i^j$ for the additive case (or $\sum_i A_{k,i} u_i^j$ under the log transform). The rank of the stacked system determines how many targets are simultaneously achievable. With n stop values and $K \cdot J$ moment-count constraints, the system is solvable when the stacked matrix has full row rank, which holds generically for $K \cdot J < n$. □

Remark 11.8 (Per-Reel and Global Volatility of the Value-Bearing Component). The per-reel variance of the value-bearing component at count k is $\mathrm{Var}(\omega|\text{count} = k)$, targetable by the distributional control above. The per-reel total value-bearing variance combines the per-count variances weighted by the count distribution:

$$\mathrm{Var}(\Omega_s) = \sum_{k=1}^{W} P(\text{count} = k) \cdot \left[\mathrm{Var}(\omega|k) + \bar{\omega}_k^2\right] - \mathbb{E}[\Omega_s]^2. \tag{62}$$

The count probabilities $P(\text{count} = k)$ are controlled by Layer 3 (τ^2). The per-count variances $\text{Var}(\omega|k)$ and means $\bar{\omega}_k$ are controlled by the linear system (Proposition 11.3). All factors are independently adjustable, so the per-reel value-bearing volatility is fully targetable. The global value-bearing volatility across reels follows from the independence of reel spins.

11.5 Expected Value of the Value-Bearing Component

Proposition 11.9 (Value-Bearing EV)**.** *For additive evaluation (cash scatters), the expected contribution of the value-bearing component on a single reel is:*

$$\mathbb{E}[\Omega_s] = \sum_{k=1}^{W} P(\mathit{count} = k) \cdot \bar{\omega}_k, \tag{63}$$

where $P(\mathit{count} = k)$ is determined by the count distribution (Layer 3, Proposition 3.8) and $\bar{\omega}_k$ (the per-count arithmetic mean) is determined by the linear system (Proposition 11.3). For multiplicative evaluation (wild multipliers), the RTP contribution requires the per-count arithmetic *mean of the product, which is degree k in the stop values and not controlled by the log-transform linear system. The log-transform targets the geometric mean; if the arithmetic mean is required, the system is polynomial and must be solved directly, though it remains solvable with $n - K$ free parameters.*

11.6 Applications

Remark 11.10 (Cash Scatters)*.* A cash scatter is a value-bearing symbol with additive evaluation. Each scatter stop carries a cash value. The scatter's total payout on a spin is the sum of visible values across all reels. The per-reel expected contribution is controlled by the linear system (Proposition 11.3). The cross-reel total is the sum of independent per-reel contributions (since reels spin independently), so the scatter's overall EV and variance are computable from the per-reel value-bearing EVs.

Remark 11.11 (Scatter Count Probabilities)*.* Unlike OAK symbols (evaluated per reel per window), scatters are typically evaluated by total count across all reels. The probability of exactly j scatters across R reels is a convolution of the per-reel count distributions. Since each reel's scatter count distribution is determined by its coverage $p^{(i)}_{\text{scatter}}$ and its count distribution, the cross-reel scatter-count probabilities are pre-computable. The probability of triggering a feature (e.g., 3+ scatters) is a known function of the per-reel coverages.

Remark 11.12 (Multiplier Wilds)*.* A multiplier wild is a value-bearing symbol with multiplicative evaluation. Each wild stop carries a multiplier value (2×, 3×, 5×, etc.). When k multiplier wilds appear in a window, the total multiplier is the product of their values. Under the log transform $u_j = \log \omega_j$, the per-count geometric mean is targeted by the same linear system (Proposition 11.3). The per-reel expected multiplier given count k is $\exp(\bar{u}_k)$, independently targetable at each count level.

Remark 11.13 (Integration with the Framework)*.* Value-bearing symbols are placed in the skeleton with per-reel targets (p, c, τ^2), exactly as non-value-bearing designed symbols. The numerical values are assigned after the skeleton is constructed, by the linear system (Proposition 11.3). The placement and the numerical values are independent in the following sense: the skeleton determines where the symbol appears and how it clusters, and the linear system (Proposition 11.3) determines what values the stops carry. The placement does not constrain the values, and the values do not affect the placement, the co-location matrix, or the hit rate.

The value-bearing EV does enter the total RTP: for cash scatters, it is an additive RTP component; for multiplier wilds, it scales the OAK payouts of the symbols the wild substitutes for. The designer specifies the per-count average targets as part of the overall specification, and the filler allocation (Section 5) accounts for the value-bearing RTP contribution when targeting the total RTP. The Full Construction Theorem (Theorem 9.4) applies with the value-bearing linear system as an additional post-skeleton step, and the value-bearing EV as a pre-computed component of the RTP target.

12 Scatters

A *scatter* symbol pays based on how many reels display it, regardless of position within each reel's window. (When scatters are placed as singles with $c = 1$, the number of reels displaying the scatter equals the total scatter count on screen. For $c > 1$, these differ; the Poisson-binomial model below counts reels, not individual symbols.) It is excluded from the ways-pay evaluation and is never substituted by wilds. This section shows that the scatter's count probabilities, RTP, and payout distribution are exactly computable, independently tunable, and fully integrated with the rest of the framework.

12.1 Count Distribution

Proposition 12.1 (Poisson-Binomial Characterization)**.** *A scatter symbol on R independent reels with per-reel presences $p_1, \ldots, p_R$ has count distribution given by the Poisson-binomial probability generating function:*

$$P(\textit{exactly } k) = [x^k] \prod_{i=1}^{R} (1 - p_i + p_i x). \tag{64}$$

The PMF is computable in $O(R^2)$ by sequential convolution. The probability of any threshold event $P(\geq n) = \sum_{k=n}^{R} P(k)$ is a sum of multilinear terms.

Proposition 12.2 (Multilinearity of Count Probabilities)**.** *$P(\textit{exactly } k)$ is multilinear in $p_1, \ldots, p_R$: each p_i appears to at most the first power. Its extremes over any box $\prod_i [l_i, u_i]$ are therefore attained at vertices. The exact achievable range of any count probability requires at most 2^R evaluations.*

12.2 Scatter RTP

Proposition 12.3 (Scatter RTP)**.** *The scatter RTP is:*

$$\mathrm{RTP}_{\mathrm{scatter}} = \sum_{k=0}^{R} P(\textit{exactly } k) \cdot \mathrm{pay}[k], \tag{65}$$

which is linear in the PMF and hence multilinear in the p_i. The total game RTP decomposes additively:

$$\mathrm{RTP}_{\mathrm{total}} = \mathrm{RTP}_{\mathrm{ways}} + \mathrm{RTP}_{\mathrm{scatter}}. \tag{66}$$

The two components are computed from disjoint symbol sets and are independently targetable.

12.3 Constrained Ranges

Proposition 12.4 (Constrained Ranges)**.** *Fixing $P(k^*) = t$ and solving for one presence $p_j = (t - q_k)/(q_{k-1} - q_k)$, where q_m is the PMF at count m with $p_j = 0$, any other count probability $P(m)$ becomes a rational function of each remaining p_i with quadratic numerator and affine denominator. The extrema of $P(m)$ over each $p_i \in [l_i, u_i]$ are attained at the two box endpoints or at the (at most 2) roots of the derivative's numerator. The constrained range of $P(m)$ at fixed $P(k^*)$ is computed by a coordinate-wise sweep: for each variable p_i, evaluate at the two box endpoints and the (at most 2) derivative roots, giving 4 candidates per variable. Sweeping all $R - 1$ remaining variables yields at most $R \cdot 4^{R-1}$ evaluations. For $R = 5$ this is 1,280 evaluations, trivially pre-computable. The result is a coordinate-wise sweep, not a global bound; the joint extremum may lie at a non-axis-aligned point. For exact global bounds, Lagrange stationarity on the multilinear constraint yields a polynomial system solvable in fixed dimension.*

12.4 Volatility Tunability

Proposition 12.5 (Scatter Variance at Fixed RTP)**.** *The scatter payout variance is:*

$$\mathrm{Var}(Y_{\mathrm{scatter}}) = \sum_k P(k) \cdot \mathrm{pay}[k]^2 - \mathrm{RTP}^2_{\mathrm{scatter}}. \tag{67}$$

At fixed scatter RTP (one multilinear constraint on R presences), asymmetric per-reel presences produce different count distribution shapes with different variances. The achievable variance range at fixed scatter RTP is a pre-computable interval, tunable by the choice of per-reel presences on the RTP iso-surface.

12.5 Full Payout Distribution

Proposition 12.6 (Scatter Distribution is Targetable)**.** *The scatter payout distribution has at most $R + 1$ atoms (one per count level $k = 0, \ldots, R$). Each atom's probability $P(k)$ is controlled by the per-reel presences via the Poisson-binomial. Each atom's value is controlled by the paytable or, for value-bearing scatters carrying cash values, by the linear system of Section 11. Both are independently targetable, so the full scatter payout distribution is exactly computable and exactly controllable.*

12.6 Integration with the Framework

Proposition 12.7 (Decomposition of Game Metrics)**.** *The complete game's metrics decompose into three sequentially decoupled systems and one exactly computable coupling term:*

1. Scatter metrics *(controlled by per-reel presences p_i alone):*
   - *Scatter hit rate: $P(\geq n) = \sum_{k \geq n} P(k)$, from the Poisson-binomial.*
   - *Scatter RTP: $\sum_k P(k) \cdot \mathrm{pay}[k]$, multilinear in the p_i.*
   - *Scatter variance: $\sum_k P(k) \cdot \mathrm{pay}[k]^2 - \mathrm{RTP}^2_{\mathrm{scatter}}$, tunable by asymmetric p_i on the RTP iso-surface.*

   No co-location matrix is involved. The scatter fires on a cross-reel count, not a within-window pattern.

2. Ways metrics *(controlled by filler counts and co-location matrix Q):*

- *Ways RTP: multilinear in filler counts (Section 5).*
- *Ways hit rate: targeted via Q off-diagonal (Section 7).*
- *Ways* $\mathrm{CV}_{\mathrm{win}}$*: targeted via clustering distribution, directed filler arrangement, and label permutation at frozen RTP and HR (Section 8).* σ^2 *derived.*

3. Total RTP*:* $\mathrm{RTP}_{\mathrm{total}} = \mathrm{RTP}_{\mathrm{ways}} + \mathrm{RTP}_{\mathrm{scatter}}$*. Additive, since the two systems evaluate disjoint symbol sets.*

4. Coupling terms *(controlled by the skeleton-filler block* $Q[\mathrm{scat}][f]$*):*

   - $P(\textit{ways win AND scatter win})$*: on reels showing scatter, one cell is occupied, reducing the effective window for paying symbols. The joint probability is exactly computable from the per-reel window content distributions, captured by* $Q[\mathrm{scat}][f]$*.*
   - $\mathrm{Cov}(Y_{\mathrm{ways}}, Y_{\mathrm{scatter}})$*: nonzero and negative (scatter presence reduces paying symbol counts). Exactly computable from the skeleton-filler co-location block.*

   Total hit rate: $P(\textit{any win}) = P(\textit{ways}) + P(\textit{scatter}) - P(\textit{both})$*, all terms computable in* $O(L)$ *from the constructed strip, without simulation.*
   Total variance: $\mathrm{Var}(Y_{\mathrm{total}}) = \mathrm{Var}(Y_{\mathrm{ways}}) + \mathrm{Var}(Y_{\mathrm{scatter}}) + 2\,\mathrm{Cov}$*, all terms computable in* $O(L)$ *from the constructed strip, without simulation.*

The Full Construction Theorem (Theorem 9.4) applies with the scatter as an additional skeleton symbol, its RTP as a pre-computed additive component, and its count probabilities as independently targetable parameters. The packing constraint (54) absorbs the scatter's stop counts into the skeleton footprint.

13 Lines Correction

The framework as developed uses *ways-pay* evaluation: a kOAK win counts the product of per-reel symbol counts in the window. In *lines-pay* games, wins are evaluated along P fixed paylines, each selecting one cell per reel. This section provides the correction factor that converts between the two models. The framework applies to any display height W and any number of reels R.

13.1 Per-Payline Evaluation

For a game with P paylines on an R-reel, height-W grid, each payline selects one row per reel. Under a uniform spin, the probability of symbol s appearing at a specific cell on reel i is $n_s^{(i)}/L_i$. The expected payout for a kOAK of s on a single payline is:

$$\mathrm{RTP}^{(s)}_{\mathrm{line},k} = v_k^{(s)} \cdot \prod_{i=1}^{k} \frac{n_s^{(i)}}{L_i} \cdot \left(1 - \frac{n_s^{(k+1)}}{L_{k+1}}\right), \quad k < R, \tag{68}$$

and the total lines RTP is P times this sum over all symbols and OAK levels. The blocker term uses reel $k+1$ only (left-anchored evaluation), matching the ways-pay convention (Definition 2.10).

13.2 The Correction Factor

Proposition 13.1 (Lines Correction)**.** *The per-symbol, per-OAK correction factor from ways to lines is:*

$$\lambda_{s,k} = \frac{P}{W^k} \cdot \frac{1 - n_s^{(k+1)}/L_{k+1}}{1 - p_s^{(k+1)}}, \tag{69}$$

where the numerator uses the per-cell absence probability (lines blocker: one cell on reel $k+1$) and the denominator uses the per-window absence probability (ways blocker: any cell in the window on reel $k+1$). The correction is per-symbol and per-OAK-level: no single scalar corrects all symbols at all depths simultaneously.

Proof. The ratio $\text{RTP}_{\text{lines}}/\text{RTP}_{\text{ways}}$ for symbol s at depth k is:

$$\frac{P \cdot (n/L)^k \cdot (1 - n/L)}{(Wn/L)^k \cdot (1 - p)} = \frac{P}{W^k} \cdot \frac{1 - n_s^{(k+1)}/L_{k+1}}{1 - p_s^{(k+1)}}. \tag{70}$$

The paying-reel terms simplify because $(n/L)^k/(Wn/L)^k = 1/W^k$. The blocker terms differ because the lines blocker checks one specific cell (n/L) while the ways blocker checks the entire window ($p = Wn/L$ under singles). □

Corollary 13.2 (All-Ways vs All-Lines)**.** *When $P = W^R$, the correction factor is $\lambda_{s,k} = W^{R-k} \cdot (1 - p/W)/(1 - p)$ for $k < R$. At $k = R$, there is no blocker reel and $\lambda = 1$: all-ways and all-lines agree. For $k < R$, $\lambda > 1$ because the per-cell blocker is weaker than the per-window blocker. The divergence is approximately W^{R-k}, reflecting that a ways game counts only the paying reels while a lines game counts the full R-cell path.*

Remark 13.3 (Scope of the Lines Correction). The correction factor $\lambda_{s,k}$ fully resolves the RTP computation for lines games: the total RTP under lines evaluation is a known, pre-computable function of the symbol counts and the payline geometry. The co-location matrix, the hit-rate targeting, and the volatility control operate on the ways-pay model. For lines games, the RTP targeting (Section 5) applies with λ as a multiplicative correction.

However, hit rate and volatility under lines evaluation are *not* resolved by λ alone. The hit rate in a lines game is the probability that at least one payline achieves a kOAK, which depends on which symbols occupy which specific rows in the window (not just which symbols are present). Paylines sharing cells create correlations that the per-window co-location matrix does not capture. The payout variance under lines evaluation similarly depends on per-payline covariances arising from shared cells. A complete lines-game theory for hit rate and volatility would require extending the co-location framework to per-cell, per-row statistics. This remains open.

14 Comparison to Prior Work

The existing literature on reel strip design consists of five optimization papers from two research groups (Bulgarian Academy of Sciences and University of Macedonia), plus one formal-verification approach (Eindhoven), all treating the strip as given or searching for one via metaheuristic optimization of a scalar or multi-criteria objective.

Balabanov et al. [2] apply a Genetic Algorithm to search for a symbol distribution achieving a target RTP, using Monte Carlo simulation as the fitness function. The search space is the set of symbol count vectors. Each candidate is evaluated by simulating 10^6 spins. The method finds approximate RTP matches but provides no feasibility guarantee, no hit-rate control, no volatility control, and no characterization of the attainable set.

Keremedchiev et al. [4] replace Monte Carlo evaluation with exact full-cycle computation, eliminating simulation variance. The search remains a Genetic Algorithm over symbol distributions targeting RTP alone. The improvement is in evaluation speed and accuracy, not in the scope of what is targeted or guaranteed.

Kamanas et al. [5] introduce Variable Neighborhood Search (VNS) with two local search operators (swap and shift), achieving the current state of the art in RTP convergence speed. The 2025 survey by the same group [6] presents VNS as the frontier of the field, with independent hit-rate and volatility control listed as open future work.

	GA [2]	**GA+Exact** [4]	**VNS** [5]	**This paper**
RTP targeting	approx	exact	exact	exact
Hit-rate targeting	no	no	no	yes
Volatility targeting	no	no	no	yes
Per-symbol control	no	no	no	yes
Existence guarantee	no	no	no	yes
Feasibility bounds	no	no	no	pre-computed
Wild symbols	unknown	unknown	unknown	linear correction
Method	search	search	search	construction
Evaluation	Monte Carlo	full cycle	full cycle	closed form

The difference is not incremental. The prior work searches for a strip matching one target, with no guarantee of success and no control over the remaining degrees of freedom. The present framework characterizes the full achievable set, targets all three global metrics simultaneously alongside per-symbol specifications, guarantees existence when targets are feasible, detects infeasibility when they are not, and constructs a realization deterministically.

Remark 14.1. Hit-rate and volatility targeting are standard commercial deliverables — game mathematicians routinely adjust reel strips to meet these targets via iterative manual tuning or proprietary tools. What the prior *academic* literature lacks is not the practice but the theory: no published work establishes constructibility with characterized attainable sets, pre-computable feasibility bounds, or existence guarantees. The contribution of this paper is the mathematical foundation, not the design goal.

The co-location matrix, the three-layer decomposition, the RTP iso-surface, the reachability of the achievable set via splice linearity, and the $E[c^2]$ lattice have no precedent in the slot-design literature.

15 Worked Example: Construction of a Complete Game

This section constructs a complete game from specifications, exercising every section of the framework. The game is a 5×3 ways-pay machine with $3^5 = 243$ ways, bet = 1 credit, and total RTP = 94% (64% base game +30% free games). Three design coordinates are targeted simultaneously: RTP = 64.00%, hit rate = 1 in 4.0, and $\mathrm{CV}_{\mathrm{win}} = 15.0$. The free-game RTP is determined by the free-game reelset and trigger frequency; the specific EV of the free-game mode depends on those strips, which are constructed independently. Changing the free-game contribution would modify the free-game strips, not the base-game strips built here.

15.1 Specification

Paytable. Ten paying symbols in two depth classes (Table 1).

Symbol	Depth	2OAK	3OAK	4OAK	5OAK
H1	2	0.40	1.00	4.00	10.00
M1	2	0.15	0.50	2.00	6.00
M2	2	0.10	0.30	1.00	4.00
M3	2	0.05	0.20	0.60	2.50
A	3	—	0.10	0.40	1.50
K	3	—	0.10	0.30	1.00
Q	3	—	0.05	0.20	0.80
J	3	—	0.05	0.15	0.60
10	3	—	0.05	0.10	0.40
9	3	—	0.05	0.10	0.30

Table 1: Paytable (pays per way per credit wagered). Depth classes: $\{H1, M1, M2, M3\}$ at $d = 2$ (pay from 2OAK), $\{A, K, Q, J, 10, 9\}$ at $d = 3$ (pay from 3OAK). Within each depth class, label permutations are HR-neutral: $4! \times 6! = 17{,}280$ assignments.

Wild. Multiplier wild on reels 2–4 only. Back-loaded: $P(\text{any wild on screen}) \approx 1/31$. Per-reel presence $p_{\text{wild}} = 0.004, 0.007, 0.022$ on reels $2, 3, 4$ respectively. Clustering: $c = 1.0$ (reel 2, singles), $c = 1.2$ (reels 3–4). A moat of $W - 1 = 2$ positions separates the wild cluster from $H1$.

Scatter. $p = 0.10$ on all five reels, placed as singles. Cash pays: 3 = \$1, 4 = \$2.50, 5 = \$10. Trigger probability $P(3{+}) = 1/117$. Cash RTP: 0.93%.

15.2 H1 Premium Symbol

Five per-OAK hit rate targets: $P(\text{2OAK}) = 1/15$, $P(\text{3OAK}) = 1/30$, $P(\text{4OAK}) = 1/250$, $P(\text{5OAK}) = 1/600$, and a near-miss target $P(\text{miss on reel 2 only}) = 1/150$ (symbol present on reels 1, 3, 4, 5 but absent from reel 2). Five equations in five unknowns (per-reel effective presence $p_1, \ldots, p_5$), solved by the ratio chain (Proposition 3.1). Per-reel clustering targets c_i set independently. The resulting skeleton specification (Table 2):

15.3 Wild Multiplier as Effective CC

The wild carries a compound multiplier: when k wild stops appear in a window, the multiplier is $\prod_{j=1}^{k} \omega_j$. By Definition 11.2, the per-count average for multiplicative evaluation is $\bar{\omega}_k = \mathbb{E}[\prod \omega_{\text{visible}} \mid \text{count} = k]$. The compound multiplier folds into the wild's effective conditional count:

$$c_{\text{eff}} = P(1) \cdot 1 \cdot \bar{\omega}_1 + P(2) \cdot 2 \cdot \bar{\omega}_2, \tag{71}$$

where $P(k)$ is the probability of count level k given presence. This c_{eff} replaces c_{wild} in the additive formula $m_{\text{eff}} = m_{\text{native}} + p_{\text{wild}} \cdot c_{\text{eff}}$ (Proposition 10.1), which sets the floor for every paying symbol's

	R1	**R2**	**R3**	**R4**	**R5**
L	511	1500	720	2067	510
p_{eff}	0.5284	0.2000	0.3681	0.1451	0.2941
c_{H1}	1.2	54/49	1.8	1.8	1.8
n_{H1}	108	108	156	153	90
p_{wild}	—	0.004	0.007	0.022	—
c_{wild}	—	1.0	1.2	1.2	—
n_{wild}	—	2	2	18	—
n_{scat}	17	50	24	69	17
n_{filler}	386	1340	538	1827	403

Table 2: Skeleton specification. Per-reel L is derived from integrality constraints: the largest L under a design cap such that $n = pcL/W$ and $a = pL$ are integer for every designed symbol on that reel. Reel 2 is concatenated $K = 2$, reel 3 $K = 2$, reel 4 $K = 3$ (Proposition 4.11): reel 2 for a minimum of 2 wild stops, reel 4 for finer distributional control over the multiplier values from $\{2, 3, 5\}$; $p_{\text{eff}} = p_{\text{native}} + p_{\text{wild}}$.

effective m on that reel. The compound across reels is handled by the standard product $\prod m_{\text{eff}}$ in the RTP formula.

At uniform ω per reel (all stops carry the same value), the per-count averages are $\bar{\omega}_1 = \omega$ and $\bar{\omega}_2 = \omega^2$. The per-reel values are solved to hit RTP = 64% at the given R4 distribution; CV_{win} emerges from the resulting compound structure. The solution uses non-uniform stop values on reel 4 from $\{2, 3, 5\}$: two stops at $\omega = 2$, five at $\omega = 3$, and eleven at $\omega = 5$. Count-2 windows on reel 4 see compounds ranging from $2 \times 3 = 6$ to $5 \times 5 = 25$ depending on which stops are visible — this dispersion drives CV_{win} above the uniform baseline. Reel 4 is concatenated $K = 3$ (Proposition 4.11) to provide 18 stops for finer distributional control; reels 2 and 4 are both concatenated, reel 2 for a minimum of 2 wild stops. Reels 2 and 3 carry uniform values $\omega_2 = 1.93$ and $\omega_3 = 2.90$, solved for RTP = 64%. The per-stop values are assigned from $\{2, 3, 5\}$; the compound multiplier in each window is computed by exact enumeration over all L_i windows per reel, not by the linear system of Section 11 (which targets per-count arithmetic means, not the arithmetic mean of products arising from multiplicative compounding). The non-uniform stop value distribution is the primary CV_{win} lever.

15.4 Filler Allocation and Scatter

Nine fillers are allocated on the RTP iso-surface via the intermediate value theorem (Proposition 5.8), with blend parameter $t = 0.19$ between uniform and pay-weighted distributions. Total filler RTP: 15.6%. Combined with $H1$ at 47.5% and scatter at 0.93%: 64.00% base game RTP.

Scatter constrained range. The Poisson-binomial structure (Proposition 12.1) constrains the achievable scatter-count distribution. At per-reel $p = 0.10$: $P(3+) = 1/117$, $P(3) = 1/123$, $P(4) = 1/2{,}222$. A target of $P(4) = 1/500$ is infeasible at this $P(3)$: the ratio $P(4)/P(3)$ is constrained by the Poisson-binomial coupling.

15.5 RTP Verification

Full enumeration over all L_i windows per reel yields exact m_{eff} and p_{eff} per symbol per reel, including the compound wild multiplier folded into the CC (Table 3).

Symbol	RTP	5OAK freq	Any win freq
H1	47.5%	1/600	1/9
M1	6.9%	1/800	1/14
M2	3.6%	1/810	1/14
M3	2.0%	1/1050	1/16
A	1.0%	1/1130	1/71
K	0.7%	1/1150	1/70
Q	0.5%	1/1080	1/70
J	0.4%	1/1220	1/72
10	0.3%	1/1200	1/75
9	0.2%	1/1200	1/73
Scatter	0.93%	—	1/117
Total	**64.000%**		

Table 3: RTP per symbol, verified by exact enumeration.

15.6 Three-Coordinate Targeting

The variance decomposition identity (Proposition 8.1) gives $\sigma^2 = (\text{RTP}^2/p)(1-p+\text{CV}_{\text{win}}{}^2)$, where p is the hit rate and CV_{win} is the conditional coefficient of variation of the win amount given a win. The three coordinates $(\text{RTP}, p, \text{CV}_{\text{win}})$ are targeted by three sequentially decoupled controls.

Hit rate structure: $H = S_1/\Omega$. The hit rate decomposes into $S_1 = \sum_s \prod_{i=1}^{d_s} p_s^{(i)}$, the expected number of winning symbols per spin (pure order 1, reads presences only), divided by $\Omega = S_1/H = \mathbb{E}[\#\text{winning symbols} \mid \text{win}]$, the overlap factor (one scalar, all co-location). On the present design at the singles operating point: $S_1 = 0.497$, $H = 0.305$, $\Omega = 1.63$, overlap cost 38.7% of the naïve disjoint sum. RTP reads m_i on paying reels and p on the blocker reel only; S_1 reads p_i on paying reels. These are disjoint: presences on non-blocker reels move S_1 (and hence H) without touching RTP.

Hit rate targeting. The achievable hit-rate range at RTP = 64% is $\sim$1/3 (all fillers as singles) to $\sim$1/10 (all fillers maximally clustered on non-blocker reels). Filler clustering trades p against c at fixed $m = p \cdot c$ (Theorem 2.7), so RTP is exactly preserved. Clustering is restricted to non-blocker reels (Corollary 2.13): depth-2 fillers may cluster on reels 1–2; depth-3 fillers on reels 1–3. Target: 1/4.0. Each filler's per-reel c_f is a separate design parameter (Definition 5.1); the per-symbol c_f values are chosen so that the resulting per-symbol presences $p_f = m_f/c_f$ yield the target HR through the inclusion–exclusion formula. The splice construction (Proposition 6.32) fills gaps between lattice points at $O(1/n)$ resolution.

HR-preserving moves and the volatility kernel. On each reel with the others fixed, H is exactly linear in the per-reel window-type counts (the full vector of W-gram frequencies, not just the pairwise co-location Q). The order-2 contribution $H^{(2)}$ is a function of Q alone (Corollary 6.19);

the order-3 terms depend on triple co-occurrence, a strictly finer statistic. At the $H^{(2)}$ level, every gradient $\partial H^{(2)}/\partial Q_r[s,t]$ is negative: more co-location always lowers H (shared windows make wins coincide). The HR-preserving moves at fixed presences form a linear subspace (the kernel of the gradient), spanned by leverage-weighted trades: increase $Q_r[s,t]$ by $\varepsilon/g_r[s,t]$ and decrease $Q_{r'}[s',t']$ by $\varepsilon/g_{r'}[s',t']$, where $g = \partial H^{(2)}/\partial Q$. Net $\Delta H^{(2)}$: exactly zero; the residual $H - H^{(2)}$ (measured at $\sim 10^{-4}$ on this design) is correctable by the splice construction. Both $H^{(2)}$ and $\mathbb{E}[Y^2]$ are linear in the co-location entries on each reel, so each pair (s,t) carries an exact *efficiency ratio*: volatility gained per unit of hit rate spent. The HR-preserving volatility trade is explicit: raise a high-efficiency pair (e.g., H1 adjacent to M1), lower a low-efficiency one (e.g., H1 adjacent to 9), net $\Delta H^{(2)} = 0$, net $\Delta\mathbb{E}[Y^2] > 0$.

$\mathrm{CV}_{\mathrm{win}}$ **targeting.** The per-reel multiplier value distribution is the coarse $\mathrm{CV}_{\mathrm{win}}$ control. At uniform ω per reel: $\mathrm{CV}_{\mathrm{win}} \approx 11$ (the baseline). Introducing non-uniform stop values on reel 4 from $\{2,3,5\}$ creates compound variance across count-2 windows. The distribution is the degree of freedom: $[0{\times}1, 2{\times}2, 2{\times}3, 14{\times}5]$ gives $\mathrm{CV}_{\mathrm{win}} \approx 14.5$; shifting stops toward $\omega = 3$ ($[0{\times}1, 2{\times}2, 5{\times}3, 11{\times}5]$) gives $\mathrm{CV}_{\mathrm{win}} = 15.0$. RTP is preserved exactly across distributions because it reads only per-count means; $\mathrm{CV}_{\mathrm{win}}$ reads per-count variances, which the distribution controls at frozen RTP. Reel 4 is concatenated $K = 3$ (Proposition 4.11) to provide 18 stops for this distributional control. The compound multiplier is computed by exact window enumeration.

The residual at fixed (RTP, HR). With counts fixed and the wild moat frozen (standard practice, $\sim$1.6% of filler cells), the diagonal $\sum_s \mathbb{E}[W_s^2]$ is exactly invariant. The entire residual freedom is in the cross terms $\sum_{s\neq t}\mathbb{E}[W_s W_t]$, which are linear in the per-reel co-location entries (on each reel with the others fixed). Under the wild-moat convention, the cross terms account for 100% of the residual variance. The $H^{(2)}$-preserving kernel moves change these cross terms at $\Delta H^{(2)} = 0$, giving $\mathrm{CV}_{\mathrm{win}}$ control at frozen RTP and $H^{(2)}$, with the order-3 residual correctable by splice.

Decomposition check. At the final operating point: $f_{\mathrm{HR}} = 0.33\%$. The game is deep in the premium regime: 99.7% of variance comes from win shape ($\mathrm{CV}_{\mathrm{win}}$), driven by the compound wild multiplier.

Remark 15.1 (Multipliers as a Volatility Lever). Value-bearing symbols such as wild multipliers provide a clean way to increase volatility without touching the hit rate. The maximum single-event exposure is bounded by the product of the maximum per-reel multiplier, the maximum per-reel count, and the top pay value. In principle a co-occurrence with the premium symbol could further increase this exposure; the co-location matrix and the 3-gram targets (Section 6) make this computable exactly.

Increasing the frequency of the maximum-exposure event raises $\mathrm{CV}_{\mathrm{win}}$ directly: the multiplier inflates the right tail of the conditional win distribution without changing symbol presence. This gives the designer a spectrum from a *minimum wild exposure win* (a single low-valued wild on the rarest reel) to the *maximum compound exposure* (all wild reels active, all at maximum value), with the frequency of each controlled by the per-reel presence p_{wild} and the Section 11 value assignment.

However, the more RTP that is concentrated in rare compound events, the less *realized RTP* the player experiences per session. A player encountering only the common portion of the pay distribution will ruin faster, even though the long-run RTP is unchanged. Past a threshold, further concentrating RTP into the extreme tail has diminishing returns: a 1,000× win and a 10,000× win produce similar subjective impact, but the latter locks ten times the RTP into an event the player is unlikely to witness.

The quantities that capture this tradeoff are not volatility metrics but *experiential* ones: expected session length at a given bankroll, probability of triggering a feature within a budget, probability of reaching a target multiple of the initial bankroll before ruin. These questions are computable from the payout distribution that the framework constructs. The $\mathrm{CV}_{\mathrm{win}}$ coordinate tells the designer where the game sits on the spectrum between frequent-modest-wins and rare-massive-wins; the experiential analysis tells the designer where it *should* sit for the target audience.

A practical illustration: in a ways game, ensuring that each reel can produce a full stack of the premium symbol (count $= W$) means the player will occasionally see a full reel of the top symbol land on one or two reels. This teases the maximum-pay event—all reels stacking simultaneously—and gives the player a visible goal. But if the per-reel probability of a full stack is q, the probability of all R reels stacking is q^R, which can be made astronomically small while keeping single-reel stacks common. The maximum-pay event contributes negligible RTP (it is too rare to matter), so nearly all RTP is concentrated in achievable wins. The game remains volatile—the $\mathrm{CV}_{\mathrm{win}}$ is still high from the compound multiplier and the moderate wins—but the teased jackpot does not starve the player's session.

15.7 Strip Construction

Each reel comprises quasi-contiguous clusters (Definition 4.1) of designed symbols (H1, wild, scatter) separated by moats of filler positions. The moat ensures that no window contains stops from two different designed symbols, so the additive formula $m_{\mathrm{eff}} = m_{\mathrm{native}} + p_{\mathrm{wild}} \cdot c_{\mathrm{eff}}$ holds exactly for designed symbols. Filler stops in the moat may share windows with the wild, so their effective contribution includes a multiplicative correction $\mathbb{E}[X_f \cdot M]$ that the additive formula understates. This is handled correctly by the exact window enumeration used for the compound multiplier (Section 11), not by the closed-form formula.

The filler arrangement is constructed by the splice method (Proposition 6.32). On each reel, two extreme-point filler arrangements are built sharing the same skeleton and counts: strip A (fillers spread as singles, maximizing H) and strip B (fillers grouped into runs, minimizing H). The target hit rate $H^* = 1/4.0$ falls between the two extremes. The splice $A^{n_A}B^{n_B}$ at ratio $n_B/(n_A+n_B)$ hits the target at resolution $O(1/n)$. For the present game, 2 copies of strip A and 1 copy of strip B ($n = 3$) achieve $H = 1/4.0$ with RTP drift 0.000pp.

The clustering distribution τ^2 of the $H1$ cluster may be chosen independently at each reel: minimizing τ^2 (uniform gaps) minimizes the marginal variance of the $H1$ paying event, while maximizing τ^2 (concentrated gaps) maximizes it. This is an additional volatility lever at frozen RTP and hit rate.

The full strips (per-reel L as in Table 2) are given in Appendix A.

Remark 15.2 (Pipeline Speed). The entire pipeline — from an arbitrary game specification to verified strips — runs in under 17 seconds:

1. *Count allocation* (RTP targeting): bisection on the filler scale factor, $\sim$60 ms.
2. *Strip construction*: skeleton placement plus filler assignment, $<$ 100 ms per reel.
3. *Hit-rate targeting*: build strips A and B, measure HR by exact inclusion–exclusion ($\sim$1 ms), compute splice ratio. Total $<$ 1 s.
4. *CV targeting*: distribution search over multiplier value assignments, each requiring an omega solve ($\sim$1 s per candidate in Q-space). $\sim$10 s for the full enumeration.
5. *Verification*: exact enumeration of all windows per reel, $<$ 1 s per reel.

Total wall time from specification to verified strips: under 17 seconds on commodity hardware. This compares to hours or days of Monte Carlo simulation in the current industry workflow, with no guarantee of convergence to the target.

15.8 Summary

Target	Value	Control	Section
RTP	64.00%	Filler allocation + multiplier solve	5, 11
Hit rate	1 in 4.0	Filler clustering + Q flow construction	2, 6, 4
$\mathrm{CV}_{\mathrm{win}}$	15.0	Stop value distribution + concatenation	8, 11
σ	19.3	Derived from decomposition	8
f_{HR}	0.33%	Premium regime	8

Table 4: Three-coordinate targeting: three sequentially decoupled controls.

The entire construction—from specification to verified strips—runs in under 17 seconds on commodity hardware. Every step is closed-form or finite enumeration. No Monte Carlo simulation is required; no iterative search is performed. The framework targets all three design coordinates simultaneously through sequentially decoupled controls that compose in sequence, each preserving what the previous established, on a game with compound wild multipliers, scatter triggers, and mixed paying depths.

16 Design Abstraction and Extensions

A natural concern is that a constructive framework, by imposing mathematical structure, restricts the designer's creative freedom. The opposite is true. When strips are constructed to order from specifications, the designer works in *experience space* rather than symbol space. The reel strip becomes a compiled artifact, not a hand-tuned one. This section illustrates the level of abstraction the framework supports.

16.1 Multi-State Games as Markov Chains

Modern slot games are multi-state: a base game, one or more free-game modes, pick bonuses, and progressive features, connected by trigger events. The game's long-run behavior is a Markov chain over states, where each state has its own reelset and the transitions are determined by trigger probabilities.

Remark 16.1 (Per-State Specification)*.* In the present framework, each game state is an independent design problem. The designer specifies, for each state:

- Per-symbol targets: (p, c, τ^2) per reel, controlling the event-level experience (how often each symbol appears, in what clusters, with what visual density).
- Global targets: (RTP, hit rate, volatility) for that state, controlling the payout profile (how much of the house edge is returned, how frequently, with what variance).

The framework constructs each state's reelset independently by Theorem 9.4. Let $\mathbb{E}[\mathrm{payout}_i]$ denote the expected payout per spin in state i, and let w_i denote the wager in state i ($w_i = 1$ for the base game, $w_i = 0$ for free-spin states). The Markov chain's stationary distribution $\boldsymbol{\pi}$ then determines every aggregate metric:

$$\mathrm{RTP}_{\mathrm{game}} = \frac{\sum_i \pi_i \cdot \mathbb{E}[\mathrm{payout}_i]}{\sum_i \pi_i \cdot w_i}, \tag{72}$$

$$H_{\mathrm{game}} = \sum_i \pi_i \cdot H_i. \tag{73}$$

The denominator sums only the wagered states; free-spin states contribute expected payout to the numerator but no wager to the denominator, correctly capturing their effect on the overall return. The overall volatility is computable from the per-state variances and the transition structure: $\mathrm{Var}_{\mathrm{game}}$ accounts for both within-state variance (per-spin variance at each state's reelset) and between-state variance (the RTP differences between states, which produce session-level variance as the game transitions between high- and low-paying modes). Both components are functions of the per-state targets and the transition matrix, all specified by the designer.

The same aggregation applies within a feature. A free-game sequence with locking wilds passes through a sequence of conditional reelsets (one per locked-wild configuration). Each conditional reelset has its own per-symbol targets and global metrics. The feature's aggregate RTP, hit rate, and volatility are computed from the sub-chain over conditional states. The designer specifies the experience at every level of the hierarchy: per-symbol within each state, per-state within each feature, per-feature within the game.

16.2 Free Games with Locking Wilds

Consider a free-games feature with locking wilds and retriggers, a common modern game structure:

1. *Trigger*: scatter symbols on the base game trigger N free spins.
2. *Locking wilds*: each wild that lands during free games remains locked in place for the remaining spins, progressively improving the reelset.
3. *Retrigger*: scatter symbols during free games add additional spins.
4. *Conditional reelsets*: as wilds lock, the effective reelset changes. Each configuration of locked wilds defines a distinct game state.

The designer specifies this feature entirely in experience space:

- Base game RTP, hit rate, volatility (the everyday experience).
- Free-game trigger frequency (e.g., 1 in 200 spins).
- Free-game per-spin RTP (typically much higher than the base game).
- Retrigger probability per free-game set.
- Expected number of wilds locked by end of session.
- Overall game RTP (the regulatory requirement, typically 88%–96%).

The framework compiles these into reelsets:

1. The overall RTP constraint determines the relationship between base-game RTP, free-game RTP, trigger frequency, average session length, and retrigger probability. This is a linear system in the per-state RTPs weighted by the Markov chain's stationary distribution [13].

2. Each per-state RTP, hit rate, and volatility target is realized by a reelset constructed via Theorem 9.4.

3. The locking-wild progression is a sequence of conditional reelsets, each differing from the previous by the addition of locked wilds at specific positions. Since wild positions are skeleton positions, each conditional reelset is a known modification of the base free-game skeleton, and the framework's per-state constructibility applies to each.

4. The scatter trigger frequency is a per-symbol coverage target (p_{scatter}), achievable by the skeleton constructibility theorem (Theorem 4.9).

The designer never touches a reel strip. The experience specification determines the mathematical targets, and the framework constructs the strips that realize them.

16.3 What the Designer Controls

The framework supports the following design axes, all independently specifiable:

- *Per-symbol hit rate*: how often each symbol appears in a window (Layer 1, coverage p).
- *Per-symbol visual density*: how many stops of each symbol appear when it hits (Layer 2, conditional count c).
- *Per-event volatility*: the spread of per-symbol counts across windows (Layer 3, count variance τ^2).
- *Near-miss frequency*: the probability of a symbol appearing on $k - 1$ of k required reels, controlled by per-reel coverage targets.
- *Total RTP*: the house edge, targeted to arbitrary precision via filler allocation.
- *Total hit rate*: the win frequency, targeted via the co-location matrix.
- *Total volatility*: the payout variance, targeted via count allocation on the RTP iso-surface and fine-tuned by clustering and co-location rotations.
- *Feature triggers*: scatter frequencies, bonus trigger rates, retrigger probabilities, all specified as per-symbol coverage targets.
- *Multi-state payout profiles*: per-state RTP, hit rate, and volatility, with the overall game RTP constrained by the Markov chain.

Each axis is independently adjustable within its pre-computable attainable range. The "feel" of the game is not lost by the mathematical framework; it is *parameterized* by it. Every design intuition ("I want the free games to feel generous but volatile," "I want near-misses on the premium symbol," "I want the base game to be tight with frequent small wins") translates into a numerical target that the framework can realize.

17 Conclusion

This paper presented the first constructive theory of reel strip design from mathematical specifications. The rearrangement invariant decomposes the per-symbol design space into three sequentially decoupled layers (coverage, conditional count, count distribution), each with a characterized attainable set. The existence and constructibility theorem guarantees that any specification within these ranges can be realized by an actual reel strip. The filler system targets total RTP via the intermediate value theorem on a convex allocation polytope. The co-location matrix captures the order-2 window statistics that control hit rate, and the achievable set in $(H, \mathbb{E}[Y^2])$ is reachable by exact splice linearity, so any interior target is reached by mixing. The full construction theorem assembles these results: for any per-symbol targets and any global targets (RTP, hit rate, volatility) in the characterized achievable set, a reel strip realizing the full specification exists and can be constructed deterministically. No simulation is required; targeted refinement replaces undirected search.

Wild symbols are handled by a linear correction to each paying symbol's effective count, with a closed-form adjustment for all-wild ways and a pre-computable degenerate window adjustment to the co-location margin laws. Symbols carrying numerical values (multiplier wilds, cash scatters) are handled by a linear system targeting per-count expected values, with concatenation providing arbitrary distributional resolution. The worked example (Section 15) constructs a complete game targeting $(\text{RTP}, \text{hit rate}, \text{CV}_{\text{win}}) = (64\%, 1/4, 15)$ with compound wild multipliers from $\{2, 3, 5\}$, demonstrating all three coordinate controls operating simultaneously. The framework supports multi-state games as Markov chains, with each state's reelset constructed independently and the aggregate metrics computed from the stationary distribution.

17.1 Open Problems

Several directions remain for future work.

Lines-game hit rate and volatility. The lines correction factor $\lambda_{s,k}$ (Section 13) resolves RTP for lines games. Hit rate and volatility under lines evaluation depend on per-cell, per-row symbol placement and on correlations between paylines sharing cells. Extending the co-location framework to capture this row-level structure is the natural next step for lines-game support.

Hold-and-spin features. Hold-and-spin mechanics ("coin" features, lightning links) involve a sequential process: symbols land, lock in place, and respins continue until no new symbols appear. The locked-symbol accumulation is a spatial point process on the reel grid, and the feature's termination condition (no new landings) creates a geometric-like distribution over session lengths. Modeling the feature's RTP, hit rate, and volatility requires extending the present framework to handle correlated, state-dependent reelsets evolving within a single feature activation.

Generalized evolving-state features. The locking-wild free-game example (Section 16) illustrates one pattern of state evolution. A general theory would treat each feature as a controlled Markov chain over reelset configurations, with the designer specifying the transition structure and per-state experience targets. The framework's per-state constructibility (Theorem 9.4) provides the foundation, but the joint optimization of transition probabilities and per-state targets to achieve aggregate feature metrics (total feature RTP, expected session volatility, retrigger dynamics) remains an open design problem.

Pair moments under a skeleton. The attainable set of pair moments $\mathbb{E}[c_s c_t]$ under a skeleton is exactly decided for free reels but open for the skeletoned case at production strip lengths. Closing this certifies the hit-rate band and bounds the cross-term share of $\mathbb{E}[Y^2]$.

17.2 Closing Remarks

The existing literature treats reel strip design as a search problem: propose a strip, evaluate it, iterate. This paper reframes it as a construction problem: specify the targets, verify feasibility, build deterministically. The shift from search to construction is made possible by the structural decomposition of the design space into functionally decoupled, characterized layers, each with existence guarantees and pre-computable bounds.

The practical implication is that the game designer's task changes from trial-and-error strip adjustment to specification of the player experience. The mathematical framework does not replace design intuition; it provides a compiler that translates intuition into guaranteed-correct implementations.

References

[1] I. S. Telnaes, "Electronic gaming device utilizing a random number generator for selecting the reel stop positions," U.S. Patent 4,448,419, May 15, 1984.

[2] T. Balabanov, I. Zankinski, and B. Shumanov, "Slot machines RTP optimization with Genetic Algorithms," in *Proc. International Conference on Numerical Analysis and Its Applications*, Springer, 2015.

[3] T. Balabanov, I. Zankinski, and B. Shumanov, "Slot Machine RTP Optimization and Symbols Wins Equalization with Discrete Differential Evolution," in *Large-Scale Scientific Computing (LSSC 2015)*, LNCS vol. 9374, pp. 210–217, Springer, 2015.

[4] D. Keremedchiev, P. Tomov, and M. Barova, "Evolutionary optimization of slot machine base game RTP," in *Proc. NAA*, Springer, 2016.

[5] P.-A. Kamanas, A. Sifaleras, and N. Samaras, "Slot Machine RTP Optimization Using Variable Neighborhood Search," *Mathematical Problems in Engineering*, vol. 2021, Article ID 8784065, 2021.

[6] A. Sifaleras, "Designing modern slot machines using probabilities, combinatorics, and optimization," invited talk, Symposium Mathematics and Applications, Faculty of Mathematics, University of Belgrade, 2025.

[7] J. F. Groote, S. van Heesch, and M. Volk, "Formal Modelling and Analysis of Slot Machines," *IEEE Transactions on Games*, vol. 17, no. 4, pp. 1070–1083, 2025. arXiv:2407.06809.

[8] N. G. de Bruijn, "A combinatorial problem," *Indagationes Mathematicae*, vol. 8, no. 4, pp. 461–467, 1946.

[9] T. van Aardenne-Ehrenfest and N. G. de Bruijn, "Circuits and trees in oriented linear graphs," *Simon Stevin*, vol. 28, pp. 203–217, 1951.

[10] L. Euler, "Solutio problematis ad geometriam situs pertinentis," *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, vol. 8, pp. 128–140, 1741.

[11] C. Hierholzer and C. Wiener, "Über die Möglichkeit, einen Linienzug ohne Wiederholung und ohne Unterbrechung zu umfahren," *Mathematische Annalen*, vol. 6, pp. 30–32, 1873.

[12] D. Bertsimas and J. N. Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientific, 1997.

[13] J. G. Kemeny and J. L. Snell, *Finite Markov Chains*, Springer-Verlag, 1976.

[14] S. N. Bernstein, "On a modification of Chebyshev's inequality and on the error in Laplace's formula," *Annals of Science of the Ukrainian SSR*, vol. 4, pp. 38–49, 1924.

[15] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, Springer-Verlag, 2003.

[16] C. Barboianu, *The Mathematics of Slots: Configurations, Combinations, Probabilities*, Infarom, 2013.

[17] R. M. Wilson, "Decompositions of complete graphs into subgraphs isomorphic to a given graph," *Congressus Numerantium*, vol. 15, pp. 647–659, 1975.

A Reel Strip Listing

Complete reel strips for the worked example of Section 15. These are the *base strips* (strip A, fillers as singles). The game strips are the 3-block splice A^2B^1 described in Section 15, where strip B shares the same skeleton and counts with fillers grouped into runs. Per-reel lengths: $L_1 = 511$ (base 511, $K = 1$), $L_2 = 1500$ (base 750, $K = 2$), $L_3 = 720$ (base 360, $K = 2$), $L_4 = 2067$ (base 689, $K = 3$), $L_5 = 510$ (base 510, $K = 1$). All base lengths ≤ 1000, derived from integrality constraints. Symbol key: `H` = H1, `1--3` = M1–M3, `A,K,Q,J,T,9` = low symbols (T = 10), `W` = wild, `S` = scatter. Wild multiplier values: R2 $\omega = 1.93$ (uniform), R3 $\omega = 2.90$ (uniform), R4 from $\{2, 3, 5\}$ ($2{\times}2$, $5{\times}3$, $11{\times}5$).

Verify for yourself. Count the symbols on each reel: they match Table 2 exactly. Compute $m = Wn/L$ for each symbol on each reel and plug into the RTP formula (equation 7): the result is 64.000%, not approximately — the arrangement cannot change it. Compute $c = Wn/a$ for H1 on each reel: the clustering matches the specification. Then compute the global statistics: the per-reel effective presences $p_s^{(i)}$ (fraction of windows containing each symbol, with wild substitution) give the per-OAK hit rates via the product formula, the total hit rate H via inclusion–exclusion, and $\mathrm{CV_{win}}$ via window enumeration with the multiplier values above. The three headline numbers — RTP = 64.0%, $H \approx 1/4$, $\mathrm{CV_{win}} \approx 15$ — are all recoverable from these strips and the paytable in Table 1, with no additional input. These strips are the complete, verifiable output of the framework.

```
=== REEL 1 (L=511) ===
  0: H K H T H J H 3 H A H 1 H A H 9 H 3 H K H J H K H 1 H K H 3 H 2 H 2 H T H J H K
 40: H J H 1 H 3 H K H 3 H 9 H Q H Q H K H T H 9 H 9 H K H J H Q H T H 2 H K H 9 H Q
 80: H Q H T H A H 2 H 3 H A H 1 H 9 H A H 1 H 9 H S H 3 H J H S Q H Q 9 H 2 2 H 3 1
120: H J J H 9 9 H 1 1 H T J H Q J H K 3 H 2 2 H J J H 3 Q H K 9 H Q 2 H 3 1 H J Q H
160: A T H T 3 H S 3 H 2 9 H 9 3 H K 1 H S Q H A A H 1 9 H Q T H 1 9 H 9 2 H A J H A
200: T H A A H Q Q H 3 J H K 9 H K T H Q K H 2 2 H 1 J H A A H 3 9 H 1 A H 3 K H S A
240: H 9 2 H Q A H 2 3 H J 3 H T 1 H Q 2 H Q 2 H Q 9 H S K H J T T 3 3 S 3 J J 1 T T
280: 3 S Q A Q K S 9 9 J J Q Q S Q Q T 3 K 1 1 Q Q J 3 Q J J J S Q Q J T 9 K A 1 A 2
320: 1 Q A K J 3 K A A A 2 Q 2 J 1 1 3 T 2 1 2 K T 3 Q K 2 2 S 9 9 Q 9 1 K K 1 S 1 3
360: 1 A 3 9 J K T T 2 T Q 2 3 Q 1 9 T J 9 J J J K T J 3 J 9 1 2 A 1 T 1 A T A 9 K 9
400: T A J 1 Q J 3 3 T Q J K A J T 1 S T T A A A T 1 2 1 A A 1 A Q 2 9 9 9 2 2 Q K K
440: 9 2 A 3 T 9 K 3 3 2 2 K K 2 J J 1 T T 1 2 2 2 T 2 A Q S K 2 2 1 1 2 3 3 S A A A
480: T 1 K K 1 T T 2 T 2 2 K K 3 9 3 S 1 1 9 9 K K 1 1 T A A 9 3 3

=== REEL 2 (L=1500, K=2) ===
  0: H 9 H 3 H 2 H 1 H 1 H J H 2 H 9 H 9 H 3 H 3 H T H 1 H J H 9 H 3 H T H Q H Q H Q
 40: H 2 H J H 2 H K H A H A H A H Q H 9 H A H 2 A H T T H 1 K H J J H 2 Q H K Q H T
 80: 2 H T T H 1 1 H A A H T Q H A 3 H T T H 3 K H 2 9 H 2 K H S 9 H 3 A H A 9 H A J
120: H 9 Q H 9 9 H J 2 H 3 J H K T H 3 9 H J S H 1 3 H J K H K K H A A H T T H Q J H
160: Q Q H 1 K H 1 K H 1 K H 9 J H A T H 1 A H T 3 H 1 2 H 3 K H T T H 9 2 H Q K H A
200: 3 H K A H Q 3 H 9 J H 2 9 H K T H 1 9 H 3 A H 3 3 H 3 K H 1 Q H S T H K K H 2 2
240: H 1 T H Q Q H J K H Q T H 9 9 H 3 1 H K K H 1 A H 2 K H S 9 H 3 3 H 1 K H 9 A H
280: A Q H J K H K Q H K 1 H 1 2 W T W 1 1 K T T 3 3 A 9 S K Q K Q Q 3 K 3 9 3 K 9 T
320: 3 3 3 K K 9 1 J A 9 2 2 T 1 A T 9 9 K 3 2 T T T 1 2 3 J J 1 9 Q T 9 3 3 2 9 S J
360: A 2 K Q 1 J A 9 9 Q Q 2 A 1 1 2 2 A A Q J 9 A 3 1 Q Q T S T 1 3 A Q 1 9 A A T T
400: Q A Q J J J 3 9 J S 9 J 2 9 A A 2 J A J 9 9 1 2 A A K 3 J 3 J J S T 2 T 1 1 A A
440: 9 J J A A 2 A J T K 2 1 3 Q 3 1 2 K 3 1 9 9 9 9 A J A S 9 2 J 3 A A Q 9 Q Q Q Q
480: J 9 A 2 9 9 9 9 9 K Q 1 Q T Q Q 3 1 1 A K 9 T 2 Q 1 A 2 3 Q 2 J K 1 9 9 2 S 3 J
520: K Q 1 T 2 Q A 2 A T 9 K 2 Q 3 3 J 3 3 S K A 1 A Q 2 2 2 9 1 1 Q Q 2 Q 2 9 1 2 J
560: A 2 J 3 A 2 J K K A 2 Q Q J 3 1 S Q K K 3 3 2 Q 9 J J 9 9 9 1 3 K K T 9 1 9 S T
```

```
 600: 9 A A 2 T S Q 1 1 K J Q A Q 2 2 Q T 1 1 2 2 2 Q J 2 3 A S A 3 1 2 1 3 2 S 9 T T
 640: 2 A K 1 1 3 1 1 T T Q J 1 1 T 1 3 K J J 9 S J 2 A A J 2 2 A T Q Q 1 A A K T T J
 680: 3 A 2 A T J 9 1 Q Q S Q 2 3 3 S 3 J J J S 2 9 K K 3 1 T J 3 1 A A J 2 K 2 2 2 9
 720: K K K J 2 2 Q J Q 9 3 2 Q A 1 A Q Q K 2 2 T K T T 1 T Q 2 9 3 1 J 3 1 J 1 Q K J
 760: K 3 3 T 9 T 9 J T 3 1 9 3 3 2 2 J J T Q 2 J A T T T 1 A T K K T J 3 2 2 A T 1 3
 800: T 3 J 2 3 Q 9 2 K 1 J 9 K K K 1 9 3 J S 9 T 3 3 A Q K Q 1 T T T 1 K Q Q Q Q Q S
 840: 1 Q Q K K 3 9 S 3 3 3 J J 1 1 1 9 1 T Q A A J J 3 3 T J S Q J 1 T J Q 2 2 A 2 2
 880: J 2 1 A J A J 1 A 3 3 J A 9 3 9 K 9 T 2 Q Q 9 Q T 3 T 2 1 1 9 3 3 3 T A 9 T 1 Q
 920: 1 A 2 T S K K K T 2 3 J 1 1 1 J T 1 9 A Q 2 1 2 2 Q S 1 A T T 2 1 A A J K T A Q
 960: A T 2 S 3 T T J T Q S K T T T 3 T A 9 Q 2 T 2 J 3 2 2 J 2 2 3 9 S K T T S Q Q A
1000: 1 9 1 1 A A J Q 9 A 9 3 K A 9 J 1 A 9 Q 9 3 K K T A K J A K S 1 T A T 9 J J 9 9
1040: A A 2 1 9 Q 3 A J 2 T 9 T T 9 2 9 J 3 3 Q J 3 9 A K 2 Q Q 3 J A 1 A 1 Q Q J T K
1080: K 1 1 T A S 9 J J J Q 3 S 9 9 2 3 3 A A 2 2 K J A 3 Q K K K K T K K A J 1 2 2 1
1120: 1 K 2 Q K A J K 1 S J T J T A Q 3 Q Q 2 2 Q Q K S 2 9 J T J K 9 S J J 9 J Q A 9
1160: K J 9 T 3 A J T 9 3 1 1 3 2 2 1 3 S 3 1 2 A K K 1 1 2 T K 1 3 9 S T J Q 3 1 1 1
1200: 1 A J 3 2 3 3 3 K T 2 Q 9 J 1 S 1 1 2 1 J 2 3 Q K K T T 2 T A T Q T 9 9 1 1 J J
1240: 3 1 1 J J T 9 J S K 2 2 K A 3 1 1 2 2 9 9 2 1 Q T J Q A A A A Q K T S 3 J J T 3
1280: J A A 9 K 2 S Q 1 1 2 S K K Q A 9 T Q 9 S 3 3 3 A 1 9 A A 3 1 Q 1 J 9 S Q Q T Q
1320: 9 1 3 9 Q 1 3 A K 9 T 2 T T K Q Q J 2 2 J K 3 1 1 Q 3 3 9 J A A Q Q K 2 K A 9 J
1360: J 2 T 1 1 1 2 S 9 K K T 9 A A K K 1 3 Q A A J Q 3 S 1 K Q 3 1 T T K Q 2 2 2 A A
1400: A T A 3 2 T K T 2 S 9 3 Q K K T 9 9 3 K 3 J 1 1 J J K T J 3 J 2 Q Q 1 1 J K J J
1440: J T T J 1 3 3 K K K Q 1 A K K 9 K K T 2 K K K K 2 K 1 S 3 9 3 3 K A 9 2 2 9 9 2
1480: 1 Q 2 1 2 9 1 Q K 1 2 A Q 2 T 3 3 T 9 9

=== REEL 3 (L=720, K=2) ===
   0: H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H
  40: H H H H H H H H H H H H H H T H J H T H 3 H T H S H J H 2 H 2 H 9 H Q H A H 2 H
  80: T H 1 H T H J H 1 H Q H 2 H K H T H 1 H 2 H T H 2 H K H 1 H A H K H 2 H 2 H A H
 120: J H 3 H 1 H 3 H 3 H A H S H T H 9 H J H 2 H 9 H 2 H K H 3 H 3 H 1 H A H Q H 3 H
 160: T H 1 H 1 H Q H 1 H 2 H A H T H 2 H T H K H 1 H T H 1 H Q H K H T H J H 9 H 9 H
 200: 3 H 3 H Q H J H S H A H T H A H A H T H T H 3 H 3 H T H 9 H A H K H 3 H 3 H Q H
 240: 9 H 9 H K H 3 H 2 H J H 1 H 2 H A H T 1 W 1 W J K 1 1 3 S 2 2 9 1 1 A 2 K K 2 Q
 280: 9 K J J J 1 9 J Q T J Q T 1 9 J 3 A 2 T 9 9 A 9 9 T A A K Q Q 9 Q 9 9 S J J 2 9
 320: 3 A A J K J Q 1 1 T T K 1 S J K S 9 3 K 3 1 S 9 T 1 Q A A 1 Q Q 1 3 1 J A A 2 S
 360: K T K K 9 1 Q Q Q 1 3 1 T A 2 9 A A 1 1 9 S A 1 3 3 2 9 S 9 1 A J K K K K J 2 T
 400: Q 1 A 2 2 Q A J J S T Q 9 2 K K 1 2 9 A 9 9 K J Q A K K 1 3 2 Q Q A A K 2 A 9 1
 440: Q J S Q J J A 9 Q Q J K A 2 J Q A A Q 1 A J T J 1 2 3 1 2 1 Q S 9 T A 1 Q K J 2
 480: 9 A Q A K T T 2 9 A 2 T 2 9 3 A 3 1 T T Q 3 K K 1 9 3 K K 2 3 Q K Q A Q 9 T J Q
 520: T 9 K 1 T T 2 3 1 Q K J J J J 3 J J T K 3 9 J J J J A Q 1 9 9 9 2 K 1 2 3 3 1 2
 560: 3 S 9 3 T 1 9 9 1 1 A A 2 9 2 3 T K J J A A S Q 3 Q 9 2 2 A A 3 T Q K A A 1 3 A
 600: 3 J J 2 T S 3 1 Q 3 J 3 2 J 2 3 J 3 T 2 J J K 1 2 K K J K 1 3 T 2 2 2 J Q 1 A 9
 640: J J Q S 1 Q Q K T S K 2 2 T T 3 3 A K S A 9 2 2 Q K T T 2 3 9 9 1 K K S 3 3 A K
 680: 1 3 Q K S Q 9 S T 2 3 3 3 9 S T T T 3 1 3 Q T K K 1 1 Q 9 T K 3 Q Q 2 9 9 9 2 Q

=== REEL 4 (L=2067, K=3) ===
   0: H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H
  40: H H H H H H H H H H H H H J H J H 2 H J H J H A H J H S H Q H K H S H 2 H J H Q
  80: H 1 H 1 H 2 H J H S H J H 9 H 3 H 1 H T H K H T H K H J H K H 3 H J H 2 H Q H T
 120: H J H K H 3 H 2 H K H K H J H K H 3 H 3 H 1 H Q H T H 9 H A H A H 3 H T H T H Q
 160: H 3 H 9 H 2 H 3 H 1 H Q H A H 1 H T H 2 H 1 H 1 H 9 H Q H Q H 3 H 9 H K H Q H 3
 200: H 9 H T H K H Q H 1 H T H 9 H A H 9 H 1 H S H 3 H 3 H 1 H J H A H A H T H 2 H T
 240: H K H Q H T H J H 1 H 9 H 3 T W A W T W A W S W 2 W 3 W Q W 9 W 3 W A T W T 2 W
 280: 3 9 W J 3 W T K W T T W K S W T 3 W T 1 J A J J Q 9 T Q 9 2 Q 3 T A J Q T 2 A A
 320: J Q T S Q 2 K S 1 2 J S 2 A J 3 Q 3 A K J J 3 2 T 2 A J 2 J K J T 1 3 K K K T 9
 360: 3 K K T 9 J T 3 Q 3 K 9 A Q Q K 9 S Q Q K J 1 J 3 T 3 2 3 9 2 9 T K Q K A A S Q
 400: 2 1 S 9 1 9 T 3 1 J 1 3 A Q A Q K J J J 2 1 2 S K A Q 9 9 2 T K 9 Q 1 Q J Q 3 A
 440: A 9 1 T 2 3 3 1 Q K 1 T 2 T 1 3 2 2 A A 9 1 A K A 2 Q 9 J 3 2 J Q Q K 3 Q 1 T Q
 480: 2 J T 3 1 A 1 3 S 9 Q K K J K 9 3 A J A Q 2 3 Q T 1 A A 2 A T Q 3 9 J T 3 Q 9 J
 520: Q 1 2 3 K J 9 Q 9 K 2 J 1 K K 3 9 A 9 2 3 K J J 1 T T 3 9 2 9 T 2 T Q Q K J 1 S
 560: T A 3 Q T 1 3 1 K J J A 1 J 3 9 A 9 K 1 Q 2 K A K 1 T K Q J 9 9 9 S 1 3 K 1 A 3
 600: 3 2 Q S 2 Q 9 T J 2 J 2 9 1 J K 2 J 1 3 1 K T T K J T Q 2 9 2 9 3 1 9 A K 1 1 2
 640: 2 T 1 1 S K 2 T T 9 1 Q A 1 J A J A 1 S 3 9 Q K A 3 A 3 K 1 1 Q K 1 T 9 Q 9 1 2
 680: 9 T S K Q 9 J 1 3 T 3 T A K J A J J 9 T K 9 T 1 T 3 Q 1 1 Q K Q 9 A S 3 T 2 2 9
 720: Q 1 9 T 1 K J 1 J S 3 K 3 K A J Q Q A 2 K 9 1 1 A 9 T 3 T 9 9 9 Q A A 1 K A Q J
 760: K Q K J 3 Q Q Q 3 2 K A T S T K 1 S T T 3 9 K 3 T A 1 T J A 2 Q J K 2 2 A Q Q 3
 800: J 2 T 3 J T 2 A A J J 2 2 K J 2 S 2 K 3 2 Q A 2 3 2 2 A J 1 A Q J K J T 9 2 2 K
 840: A Q J 1 K J 9 9 Q K 3 A 9 3 Q 3 J 3 1 A 2 K 2 J J J 2 A 1 9 3 K 1 S A 2 2 Q 1 J
 880: 3 1 9 A 2 J 1 Q J Q S J 3 3 2 9 T S 3 T 2 T 2 A Q 1 Q 9 S T 9 3 1 K 3 T J T 3 T
 920: 1 A 2 9 3 3 J T 2 3 2 S 2 9 T 9 3 9 2 2 1 J A J A K 1 2 9 J K A 1 Q 1 2 A Q 1 Q
 960: J A T K 2 2 1 A J 1 9 Q J T 9 T J 3 A 9 3 K A K K T A K 9 9 J 2 1 A 2 J 9 K Q 3
1000: 1 T S Q Q T T Q Q 3 S 2 T 9 9 9 9 3 2 J S A 9 T 1 9 T 9 K S 1 1 9 J 9 T 2 T J J
1040: 2 A J 1 K 3 A 3 J T 3 3 J A 1 A 3 Q 1 T K J A 2 Q J 9 T 2 T 1 1 T T J Q Q 2 1 A
1080: A 9 A 3 9 T A T Q 1 S T 9 K J T A A K 3 T T 2 A 1 9 J Q 2 T 9 1 1 3 3 J 9 3 J 3
1120: Q 1 T S 3 9 9 2 2 2 A T S 9 2 3 2 2 T S T 9 A A A A 1 3 9 3 3 Q 3 2 S T 9 1 J 9
1160: 2 T 3 A K 2 K Q 1 3 T Q J S 1 2 K T A Q 2 1 2 T J T 1 J J K 3 Q 3 A Q 9 3 A 1 1
1200: 2 Q J 9 A 3 1 Q 3 1 T S 2 2 2 T K T J K A 1 9 J Q Q Q J 9 T 1 J 9 2 3 1 A A 9 A
1240: 3 J 2 1 9 J J J S Q J Q Q Q J Q S 9 1 Q 2 S 2 A 2 1 2 S 3 J Q 3 A 3 K K 2 J 3 3
1280: J J 2 T 2 2 9 T Q 1 A J 2 Q K 1 T A 3 A Q K K 2 A 3 Q J Q T J 3 T K 1 Q 3 J A 1
1320: 3 1 S K Q J 9 1 3 Q A 2 3 2 9 A Q 3 2 A A Q T 1 9 2 T T 1 2 9 A Q Q T 2 A 2 A 1
1360: J A J 3 J 2 9 T K 2 Q 3 K S K 3 J Q J J 1 9 3 9 Q 2 9 9 Q Q J 3 2 9 9 1 3 2 J Q
1400: S Q 2 A Q 9 9 A 1 J 2 J 2 A 9 A 9 2 S T J J 9 A K Q 3 9 2 J 2 Q Q J J Q J 3 9 S
1440: 1 T 1 A A Q K K 2 2 K A A 2 Q J Q Q K J A T Q T J A 1 A T 1 3 3 K A K 1 K S J K
1480: 1 1 T 1 A 3 3 1 A K 9 K K 1 K K 1 Q J 2 Q 2 K Q K K 3 J Q K 3 3 T 2 2 1 Q 3 2 1
1520: 2 J K 1 A A J K A A 2 2 J K 3 1 3 1 1 J T A 3 Q 2 K K K K 1 2 1 A 9 K S 9 K J 9
1560: J 2 J 9 3 3 Q Q J 1 1 K T 9 1 1 2 A 1 9 2 A Q 3 1 A 9 2 1 J K Q K 2 Q 3 9 1 3 K
1600: K A K 1 1 9 A K 9 Q K K T 3 1 1 S Q 1 T A 3 K 1 A Q K Q 2 2 A Q 1 2 T T T 1 9 J
1640: 2 9 1 T S 9 3 S K 3 K A 1 T 1 A 1 S 1 9 K 1 K J Q J A 1 T K J 2 1 2 K 3 9 3 9 1
1680: 1 K 9 3 J 3 T J K T A 3 A 3 9 2 1 A Q Q Q 9 A 1 S K 9 J K K A 1 K Q S A 3 3 2 A
1720: A K T Q K Q J 1 K Q A K 2 1 T 1 J 3 1 9 1 9 2 1 3 Q 2 3 A K 1 A A 9 T 9 2 K A 1
1760: J K 2 K 9 1 Q K T 1 1 9 K K T 1 A 3 K Q K 9 9 A 2 J 3 T Q 2 J A 1 1 3 T T K T 2
1800: Q 2 9 K A J 3 1 1 J 1 K Q Q S K Q 1 T K A T J 1 2 T T A T 9 2 2 T 3 9 S T T 2 2
1840: Q K 1 K A A J A T K 3 K 3 K K 9 9 Q 9 9 1 3 S T A Q 3 2 1 3 Q 2 S T 9 J 3 T 2 9
```

```
1880: 2 2 A 2 A 9 3 1 2 A 2 T Q A Q T T 2 3 J 3 S T 9 J K 3 J Q A T 1 T 9 T J 2 A 3 T
1920: S J 1 Q Q 1 3 1 K 2 A K A 3 9 1 9 J 3 9 9 9 J 9 9 Q 9 T S Q T T 3 K T A T 2 1 K
1960: S K 1 Q K Q 3 2 2 2 K 3 K K 3 T A J A S T 3 A A J 3 T 9 1 1 A T 1 S 9 3 1 T 9 K
2000: 9 9 1 2 3 J 2 1 K 2 K T Q 3 T K 9 Q 1 K K 9 3 9 K T K 2 T 3 T 2 A A 2 2 9 Q Q Q
2040: Q A 1 J S 9 2 A 2 9 9 1 3 A 9 2 A 3 Q 9 2 A J 1 J 1 3

=== REEL 5 (L=510) ===
   0: H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H H Q H K H 1 H 9 H
  40: K H A H K H 3 H 3 H 9 H J H 2 H A H 3 H 2 H A H 2 H Q H A H 1 H 3 H T H T H 2 H
  80: 9 H 9 H T H 9 H Q H T H K H K H 1 H 9 H K H A H S H 3 H 2 H A H 1 H 2 H 3 H K H
 120: 3 H K H Q H S H 3 H Q H 2 H J H 3 H 2 H K H T H S H J H 2 Q 1 J K Q K 2 1 K 2 K
 160: 1 Q K K 3 Q Q 3 1 A Q A J 3 A 9 T 3 Q 2 Q J J Q A T 9 T A J J J 2 3 Q Q S 9 Q 9
 200: 1 T A T K 2 2 3 T T 1 3 9 9 3 A 9 K A 3 9 9 9 J K 3 T 3 Q A T T K J 2 K T 2 Q 3
 240: Q K J K 1 J 3 A T J Q 3 1 1 1 A 1 2 3 T J T A J J K T 3 1 3 2 A T 1 A A A 9 T T
 280: 1 3 3 2 T A Q Q Q 3 A J A S 1 J 9 T J S 1 2 3 S 1 K 2 J S 1 2 2 K 9 3 9 J J 1 K
 320: 1 3 A K 1 T T A Q T T T K J T 2 9 T K 1 3 1 K J 2 Q S 9 3 K 3 3 3 A 1 J 1 1 2 9
 360: A 2 T 9 Q T A 2 J 1 Q T A 2 1 2 9 2 1 T Q 9 A J S Q 2 2 Q Q 1 J T 1 9 1 Q 1 A Q
 400: 2 T Q J Q J 1 A 2 9 J T 9 1 3 3 9 S 2 J J S K Q A T S 2 K Q J 9 K 1 A 9 1 3 2 9
 440: 2 9 A 1 9 9 3 J 9 Q S 1 9 9 T 3 A Q K K Q 1 2 3 1 K K K J 2 Q 9 S J A 1 K K A Q
 480: 3 J 9 S Q 9 K 2 A 2 9 3 1 A J J T K K T J A J 1 2 S A 2 2 1
```