Hypernormal Cinema

Individual Editions and the Limits of Correction

Flyxion
Independent Researcher

November 2026

Contents

A Note on the Title **1**

I Foundations of Variation **3**

1 The Space of Possible Films **5**
1.1 Why Count Editions . . . 5
1.2 The Exact Count . . . 7
1.3 Enumeration, Recognition, and Realization . . . 8
1.4 One Action in One Scene . . . 9
1.5 Three Local Productions . . . 12
1.6 A Reported Error and Its Footprint . . . 13
1.7 When a Repair Collides . . . 15
1.8 Editions Across Time . . . 19
1.9 The Lifetime Repertoire . . . 20
1.10 Questions the Example Opens . . . 22

2 Perceptual Equivalence and the Measurement of Difference **25**
2.1 What Exact Equivalence Assumed . . . 25
2.2 Three Notions of Difference . . . 26
2.3 What an Experiment Would Have to Measure . . . 27
2.4 Thresholds Are Not Equivalences . . . 28
2.5 Packing and Covering . . . 28
2.6 Across Productions . . . 30
2.7 What the Formalism Permits . . . 30

3 Histories, Folds, and What a Scene Commits To **33**
3.1 The Engine Scene . . . 33
3.2 Annotating Histories with Dramatic Function . . . 34
3.3 What a Scene Commits To . . . 35
3.4 A Correction Cannot Delete the Evidence . . . 35
3.5 Noise Overcapture . . . 36

3.6 What the Chapter Gives the Others . . . 37

II The Structure of an Edition 39

4 Slots, Takes, and the Graph of Editions 41
4.1 Constraint Structures . . . 41
4.2 Counting by Frontier Tables . . . 42
4.3 A Constraint Between Nonadjacent Shots . . . 44
4.4 Width . . . 45
4.5 Counting and Unranking at Cost Set by Width . . . 48
4.6 What the Chapter Establishes and Leaves Open . . . 49

5 Continuity Beyond Neighbors 51
5.1 Entities, Events, and Showings . . . 51
5.2 Absolute and Relative Continuity . . . 52
5.3 From a Defect to the Slots It Touches . . . 53
5.4 Levels of Amendment . . . 55
5.5 Coupled Entities . . . 55
5.6 What the Chapter Gives the Others . . . 56

6 Sound, Dialogue, and Time 57
6.1 Timing as a Second Kind of Choice . . . 57
6.2 The Acoustic Occupation of an Existing Film . . . 59
6.3 Tolerance of a Cut to a Shift . . . 59
6.4 Saturating Responses and Asymmetric Tolerance . . . 60
6.5 What the Chapter Gives the Others . . . 61

III Correction 63

7 Reports and Adjudication 65
7.1 What the Ledger Holds . . . 65
7.2 The Verdict . . . 66
7.3 Assigned Versus Played . . . 67
7.4 A Taxonomy of Defects . . . 67
7.5 Many Reports About One Slot . . . 68
7.6 What Must Be Retained . . . 68
7.7 What the Chapter Gives the Others . . . 69

8 Amendments, Footprints, and Correction Events 71
8.1 From One Instance to a Rule . . . 71
8.2 The Order in Which Corrections Arrive . . . 72

8.3 The Correction Event . . . 73
8.4 Replacement Footage . . . 74
8.5 Relaxing a Rule That the History Permits . . . 75
8.6 One Process . . . 75

9 Capacity and Robustness **77**
9.1 Capacity and Spare Capacity . . . 77
9.2 Amendments as Specified Changes . . . 78
9.3 Survival at Zero Spare Capacity . . . 80
9.4 Survival Under a Separation Threshold . . . 81
9.5 Robust Capacity . . . 83
9.6 Replacement Footage as Reserve . . . 84
9.7 Capacity Is Not Repairability . . . 84

10 Allocation Under Change **87**
10.1 Allocations, Disturbance, and Budgets . . . 87
10.2 The Exact Condition . . . 88
10.3 The Shared Example . . . 89
10.4 Sequences of Amendments . . . 90
10.5 Reproducibility . . . 91
10.6 What the Two Chapters Give Together . . . 92

IV Release and Repertoire **93**

11 Locales and the Worldwide Premiere **95**
11.1 One Specification, Many Productions . . . 95
11.2 Exact Inequivalence Across Locales . . . 96
11.3 Perceptual Separation Across Locales . . . 96
11.4 Passages Where Editions May Coincide . . . 97
11.5 The Footage a Locale Must Produce . . . 98
11.6 What the Chapter Establishes and What It Assumes . . . 99

12 Ledgers, Archives, and the Release Slider **101**
12.1 The Ledger and Its Records . . . 101
12.2 What an Archive Retains . . . 102
12.3 Privacy in a Ledger of Unique Editions . . . 103
12.4 The Release Slider . . . 104
12.5 What the Chapter Establishes and What It Assumes . . . 106

13 The Lifetime Repertoire **107**
13.1 The Budget . . . 107
13.2 Revisits and the Novelty Horizon . . . 108

13.3 Selecting Without Enumerating . . . 109
13.4 Scheduling a Repertoire . . . 110
13.5 What a Repertoire Must Guarantee . . . 110
13.6 What the Chapter Establishes and What It Assumes . . . 111

14 Sequence and Reinterpretation 113
14.1 Films as Commitments . . . 113
14.2 Retrospective Reinterpretation . . . 114
14.3 Order and the Path . . . 115
14.4 Contradiction and the Latest Film . . . 116
14.5 Repertoires as Structures . . . 116
14.6 The Tenth Year and the First . . . 117
14.7 What the Chapter Establishes and What It Assumes . . . 117

V Boundaries and Cases 119

15 Exclusion and Reference Corpora 121
15.1 A Policy and Its Three Parts . . . 121
15.2 Exact Subtraction . . . 122
15.3 Approximate Subtraction . . . 122
15.4 The Representation Decides . . . 123
15.5 Exclusion and the Packing Number . . . 123
15.6 Grammar Against Combination . . . 124
15.7 The Count Apart From the Law . . . 125
15.8 What the Chapter Establishes and What It Assumes . . . 125

16 Generation Under Constraint 127
16.1 Support, Rules, and Correction . . . 127
16.2 Rejection . . . 128
16.3 Local Constraints Without Lookahead . . . 128
16.4 Exact Sampling by Table . . . 129
16.5 Generating a Distinguishable Audience . . . 130
16.6 Hierarchical Direction . . . 130
16.7 What the Chapter Establishes and What It Assumes . . . 131

17 Case Studies 133
17.1 Study One: The Engine Scene in Three Locales . . . 133
17.2 Study Two: A Correction Through a Ledger . . . 135
17.3 Study Three: A Correction to the Record of a Measured Passage . . . 136
17.4 What the Studies Show and What They Assume . . . 138

18 Open Problems 139

18.1 Mathematical Problems . . . 139
18.2 Empirical Problems . . . 141
18.3 What Each Would Unlock . . . 143
18.4 What the Chapter Establishes and What It Assumes . . . 144

A Counting and Matching **145**
A.1 Transfer Matrices . . . 145
A.2 Ranking and Unranking . . . 146
A.3 Hall's Theorem and Its Worldwide Form . . . 146
A.4 Minimum-Cost Assignment . . . 147
A.5 Packing and Covering . . . 148
A.6 What Was Checked . . . 149

B Verification Scripts **151**
B.1 Running the Scripts . . . 151
B.2 The Scripts and Their Instances . . . 152
B.3 What a Run Establishes . . . 154
B.4 What the Appendix Assumes . . . 154

C Notation and Relation to the Earlier Volume **155**
C.1 Notation of This Book . . . 155
C.2 Correspondence of Objects . . . 157
C.3 Different Words for the Same Thing . . . 159
C.4 What the Appendix Establishes and What It Assumes . . . 160

Bibliography **161**

A Note on the Title

The word *hypernormal* is borrowed, with some distortion, from Alexei Yurchak's account of the last decades of the Soviet Union [18], in which the forms of public life became so fixed and so faithfully reproduced that nobody could say what they were for, and from Adam Curtis's use of the term as the title of a documentary [19]. This book is not about either. It is about counting, and it uses the word for one narrow property of an edition. The book does not argue that hypernormality is a cultural condition or a description of contemporary media, and it does not contest the literature the word comes from. The two concerns meet at one point only: when editions differ, what viewers have in common to talk about is thinner than it looks, as the occasion for talk of Section 1.1.2 and the worldwide premiere of Chapter 11 show, and the book counts how much thinner.

An edition is *hypernormal* here when it satisfies every rule in force and shows the viewer no seam of its assembly: nothing in it reads as a variant, a splice, or a substitution. What the story shows is a separate matter. A character may miss a strip of lawn, and that mistake is part of the film, something it may need the viewer to see. The grass that was mown and then stands uncut again is the other kind, a defect of assembly that a viewer may notice and report. The lawn scene of Chapter 1 is the standard case. Each viewer sees a lawn mown in a plausible order, with the machine travelling in a consistent direction across the cuts, and nothing about it looks like a variant. Across the audience, however, no two viewers hold the same edition, and the production has chosen on purpose to keep it so. Normality is therefore a property that each edition has on its own, and sameness is a property that no viewer can check, because checking requires access to someone else's film.

The title joins this to the second half of the question the book asks, which is how much variation a production can sustain while remaining open to correction. Correction in these chapters begins with a report, and a report begins with a viewer who notices a defect of assembly. A hypernormal edition hides its seams from every viewer, so the errors that matter most are the ones the rules did not anticipate and that each viewer sees as ordinary. A viewer who notices a character's mistake has noticed the film, and a viewer who notices the grass has noticed the assembly, and only the second kind of noticing starts a correction. The ledger of Chapter 12 and the adjudication of Chapter 7 let a report, once it is made, be traced to the viewers who hold an affected edition. They do not make an unnoticed defect noticed. That limit is what the second half of the title names, and Chapter 18 treats how corrections arrive as an empirical question the mathematics leaves open.

Two qualifications apply. The word describes the models of the chapters and says nothing about any real audience, since no experiment in this book measures what viewers notice. And the satire, to the extent there is any, is meant to be carried by the arithmetic: the counts are exact on the instances given, and the reader is invited to decide what they show.

Part I

Foundations of Variation

Chapter 1

The Space of Possible Films

A film of fixed duration and resolution is a finite array of bits, and the number of such arrays can be written down exactly. The number is of no use for deciding which arrays are films, and it says nothing about how a particular audience should receive one. This chapter starts from the count and uses it to separate three problems that are easily run together: enumerating films, recognizing films, and realizing a film for a particular viewer. Only the third problem yields counts that are both meaningful and computable, because it counts structured choices made while depicting a fixed course of events, and does not count bit strings.

The chapter then follows one ordinary action, the mowing of a lawn, through a small scene, through three local productions that each film the scene differently, and through a viewer's report that mown grass has reappeared after a cut. The example is small enough to count exactly. It is also large enough to show how a correction spreads through the shots that depend on it, why two corrected editions can collide, and what a promise of one edition per viewer can and cannot mean. Its sharpest result is a production that can give two viewers distinct editions at the premiere and, after a valid correction, can no longer keep that promise. The result gives the book a question to carry past the opening count: how much variation can a production sustain while remaining open to correction? Each of the later chapters of the book is meant to grow from a problem that the example makes concrete, and the final section of this chapter lists them.

The notation follows the earlier volume, *Generative Film: Categories, Sheaves, and Types for Cinema* [1], where it applies, and every concept used here is defined again in the text.

1.1 Why Count Editions

The book counts distinguishable editions of a film because the count answers two questions that a production has to face, one about storage and one about an audience. The storage question is how much distinguishable choice a given amount of footage, rules, and metadata affords. The audience question is what viewers who hold different editions can still say to each other. The first gives the chapters their counting, and the second is the reason anyone

would want the counting to serve a shared occasion at all. Each comes with qualifications, which are stated beside the results that need them.

1.1.1 A Rate for Production Designs

Suppose that two editions are perceptually equivalent exactly when they have the same class tuple, the assumption of slotwise exactness that Section 1.4 states. Let K be the number of distinguishable editions that a production can supply under its rules, and let B be the storage needed to reconstruct them, which is the footage, the rules, and the metadata that assemble an edition from it. The proposed rate of the production is

$$\rho \;=\; \frac{\log_2 K}{B}.$$

The numerator is the number of bits of distinguishable choice that the production affords: a viewer who is given one of K equally available editions has received $\log_2 K$ bits of selection, and no more. The denominator is the price. Two designs with the same K differ in rate if one stores the footage more cheaply, and two designs with the same B differ if one makes more of its variation perceptible. This chapter does not compute B, which depends on codecs, on how takes share assets, and on the size of the rules, and the rate is offered as a way to compare production designs once those costs and the perceptual distinctions are known. Measuring B for one design supplies a rate for that design. To show that a design improves on another requires a baseline, measured by the same method, at the same revision and the same perceptual criterion.

Production C, which the chapter reaches in Section 1.5, shows why the logarithm is taken of the perceptual count and not of the tuple count. After the agreement rule it has 102,400,000 take tuples, which is about 26.6 bits, and exactly one perceptual edition, which is zero bits of choice. It stores a hundred million ways of assembling the film and affords the viewer none. Productions A and B afford about 7.9 and 6.6 bits after correction from far smaller tuple counts, as in Table 1.1.

Production	K after correction	$\log_2 K$	Take tuples	$\log_2$ of take tuples
A	238	7.9	57,834	15.8
B	96	6.6	3,072	11.6
C	1	0.0	102,400,000	26.6

Table 1.1: Bits of perceptual choice and bits in the take tuples, after the agreement rule of Section 1.5. The take-tuple column measures the storage of variation, and the perceptual column measures what a viewer can experience.

Footage can be bought as a rate improvement, and the accounting is exact for part of it. If a production adds to a slot a new class that behaves like an existing one under the rules, the number of take tuples grows by exactly the number of admissible tuples through the class

it copies. The credit to K is different, because a copy adds a class tuple only when the new class is perceptibly distinct from the old one, and the credit to the rate also depends on the storage the copy costs. Cloning the one admissible class of production C at its last slot raises K from 1 to 2, which is one bit, at the cost of one more class of footage. Chapter 9 treats the cloning count in general and asks the further question that corrections raise: how much of the rate survives the corrections that the production can anticipate.

1.1.2 An Occasion for Talk

The audience question can be put as an exchange. Take a variant of the lawn scene in which the script has the mower leave the middle strip standing. One viewer, in an edition that uses a close take, saw the strip left uncut and says so. The other, in an edition that cuts to a wide shot of the finished lawn, did not see the moment and inferred it from the standing strip at the end. They can say, truthfully and of the same film, that the character missed a strip. They can then disagree about how obvious it was, the first holding that it was plain and the second that it was easy to miss, and each has good grounds, because each judged from the evidence that the edition provided. The model of this chapter does not vary framing, so the exchange is told in words and not counted.

The exchange has three parts, and each sets a requirement for the later chapters. There is a shared event, the script history in which the strip was missed, and it is what both viewers refer to. There is a difference in the evidence, which gives them something to compare and lets a conversation exist at all. And there is an occasion: a simultaneous premiere supplies a moment at which many viewers have just seen the film and want to talk about it. Chapters 11 and 14 ask what has to survive across editions for that conversation to work, which events must be common to every edition, which evidence may differ, and what opportunities for reinterpretation a later viewer must still have. The answers set the amount of variation that a society can tolerate in what its members watch.

1.1.3 What Is Assumed

The qualifications belong next to the claims they limit. The counts of this chapter assume exact slotwise equivalence, so that two editions are the same to a viewer if and only if their class tuples are the same. The thresholds that would replace exactness with a measured tolerance, the ε of the packing results, have not been measured for any audience. The rate ρ is a quantity for comparing designs, and it does not show that any production reaches a particular value or that a viewer values distinguishability at all. The rest of the chapter takes the counts as given and asks what a correction does to them.

1.2 The Exact Count

Consider a silent, uncompressed film of ninety minutes at twenty-four frames per second, with frames of 1920×1080 pixels and 24 bits per pixel. The film has $90 \cdot 60 \cdot 24 = 129{,}600$

frames. Each frame has $1920 \cdot 1080 = 2{,}073{,}600$ pixels and therefore $2{,}073{,}600 \cdot 24 = 49{,}766{,}400$ bits. The whole stream has

$$N_0 \;=\; 129{,}600 \cdot 49{,}766{,}400 \;=\; 6{,}449{,}725{,}440{,}000$$

bits. For a bit length N write $\Sigma_N = \{0,1\}^N$ for the set of bit strings of length N, read as streams of pixel values in a fixed order.

Theorem 1.1 (Enumeration)**.** *The map $\nu_N : \{0, 1, \ldots, 2^N - 1\} \to \Sigma_N$ sending k to its binary expansion, padded on the left with zeros to length N, is a bijection. In particular $|\Sigma_{N_0}| = 2^{N_0}$, a number with* 1,941,560,821,238 *decimal digits. If formats are specified by finite descriptions and streams in different formats are counted as different objects, the set of all streams of all formats is countably infinite.*

Proof. Every integer in $[0, 2^N)$ has a unique binary expansion of at most N digits, so padding gives a well-defined injection. Every string in Σ_N is the expansion of the integer it denotes, and that integer lies in $[0, 2^N)$, so the map is surjective. The digit count is $\lfloor N_0 \log_{10} 2 \rfloor + 1$, which evaluates to the stated value. For the last claim, the set of formats is a subset of the finite strings over a finite alphabet and is therefore countable, each format has a finite set of streams, and a countable union of finite sets is countable. The union is infinite because duration is unbounded and so N is unbounded. □

The theorem gives every film expressible in the format a name, namely an integer, and counting through the integers realizes the list. It does not make the list usable. The decimal expansion of the count runs to about 1.94 trillion digits, and nothing is learned about cinema from the fact that a film is among them. The next section asks what could be learned.

1.3 Enumeration, Recognition, and Realization

Enumeration. The enumeration problem asks for a canonical indexing of all objects in a class. Theorem 1.1 solves it for streams of a fixed format. The solution is complete and uninformative: it is indifferent to whether a stream shows a lawn, noise, or a silent frame repeated for ninety minutes.

Recognition. The recognition problem asks which streams a viewer would take to be a film. A *situation* θ bundles a viewer or population, a task, and a context, as in the treatment of perceptual comparison in the earlier volume. A *recognition criterion* is a function $\mathrm{Rec}_\theta : \Sigma_N \to \{0,1\}$, and the recognized films are $\mathrm{Rec}_\theta^{-1}(1)$. Three requirements are usually folded into such a criterion. The stream must be continuous, in the sense that nearby frames agree up to the motion they depict. This requirement is stated for frames inside a shot. Adjacent frames can disagree because a film cuts, so continuity across a cut is a different requirement, and the intelligibility of a deliberate discontinuity is a third. The criterion here is deliberately coarse and keeps the three folded together. What it shows must be intelligible, in the sense

that its images can be identified as objects and actions. And it must carry consequence, in the sense that what happens at one point bears on what can be understood at a later point. The proportion $|\mathrm{Rec}_\theta^{-1}(1)|/2^N$ is a property of the chosen criterion and of nothing else. The format does not determine it, and this chapter assigns it no value.

Realization. The realization problem starts from something already recognized. A *story world* consists of a set E of events, a prefix-closed set $\mathcal{A} \subseteq E^*$ of admissible histories, a fold $M : \mathcal{A} \to Q$ assigning each history its current state, and an appearance map $\mathsf{R} : Q \to \mathcal{I}$, as in the earlier volume's separation of history, state, and appearance. A history is authoritative once recorded, the state is derived from it, and the appearance is derived from the state and can be regenerated without changing what happened. A *work* is a script history $h^* \in \mathcal{A}$ together with the requirements that any depiction of it must satisfy. A production *realizes* the work by assembling a version that depicts h^* and meets those requirements. Many assemblies depict the same history. The realization problem asks how many there are, which of them a viewer can tell apart, and what must stay fixed across all of them.

Counting enters here for the first time with substance. The number of realizations of a scene is the number of ways to make the choices that the production leaves open, taken over a structure that fixes everything else. That structure is a small finite object, which makes the count a computation. The remainder of the chapter carries out such a count for one scene, and Section 1.7 states the question it is meant to answer.

Remark 1.2 (Exclusion). A production may declare material that its editions must not reproduce. Removing exact assets by digest is a set subtraction, and its effect on the count is exactly the removed number. Removing approximate matches depends on a dissimilarity and a threshold, removes a neighborhood around each excluded item, and interacts with the packing counts of Section 1.7. Whether a given edition is permitted under copyright law is a separate determination from whether it survives a declared exclusion policy. The topic returns in Section 1.10.

1.4 One Action in One Scene

1.4.1 The story world

Let S be a finite set of *strips*, the parallel swaths of a lawn that a mower clears in single passes. The events are passes, $E = \{p_s : s \in S\}$, and the admissible histories are the words over E in which no pass is repeated. This set is prefix-closed. The state of the lawn is the set of strips already mown, so $Q = 2^S$ and $M(h)$ is the set of strips whose passes occur in h. The appearance of a strip in state q is mown if the strip belongs to q and unmown otherwise. We use *mown* and *unmown* for the grass and reserve *cut* for the editing operation.

The script is the history $h^* = p_a\, p_b\, p_c$ on three strips $S = \{a, b, c\}$. Write $h^*_{\le t}$ for its first t letters.

1.4.2 Slots and required appearance

Definition 1.3 (Scene plan)**.** A *scene plan* for the script h^* consists of m *slots*. Slot j has a story time $t_j \in \{0, \dots, |h^*|\}$ and a set $V_j \subseteq S$ of strips in frame. Slots are listed in screen order and $t_1 \le t_2 \le \cdots \le t_m$. The *required appearance* at slot j is the function $r_j : V_j \to \{\mathsf{mown}, \mathsf{unmown}\}$ with $r_j(s) = \mathsf{mown}$ if and only if $s \in M(h^*_{\le t_j})$.

The restriction to nondecreasing story time rules out flashback, and the book returns to it in later chapters. For the example the plan has five slots, listed in Table 1.2. The first is an establishing view of the unmown lawn. The second and third follow the first and second passes. The fourth is a close view of the last strip, still unmown, and the fifth is a view of the finished lawn.

Slot j	Story time t_j	In frame V_j	Required appearance r_j	Content
1	0	$\{a, b, c\}$	all unmown	establishing view
2	1	$\{a\}$	a mown	after the first pass
3	2	$\{a, b\}$	a, b mown	after the second pass
4	2	$\{c\}$	c unmown	close view of the last strip
5	3	$\{a, b, c\}$	all mown	finished lawn

Table 1.2: The scene plan for the script $p_a\, p_b\, p_c$.

1.4.3 Takes, pools, and editions

Definition 1.4 (Takes and editions)**.** A *production* p supplies for each slot j a finite *pool* T_j^p of *takes*. A take $\tau \in T_j^p$ carries three attributes. Its *displayed appearance* is a function $a_\tau : V_j \to \{\mathsf{mown}, \mathsf{unmown}\}$, which records what the footage shows on each strip in frame. Its *screen direction* is $d_\tau \in \{L, R, N\}$, where N marks a neutral take whose direction of motion is not established. Its *perceptual class* $[\tau]$ is such that takes of the same class cannot be told apart by a viewer in the situation under consideration, although the takes may be distinct files, as with re-exports or regenerated grain. An *edition* is a tuple $e = (\tau_1, \dots, \tau_m)$ with $\tau_j \in T_j^p$. Its *class* is the tuple $([\tau_1], \dots, [\tau_m])$.

A take is *stale* at slot j if $a_\tau \neq r_j$, that is, if it shows some strip in an appearance that disagrees with the state the script prescribes at that point. Direction and displayed appearance are properties of the class, since takes in one class are indistinguishable.

Assumption 1.5 (Slotwise exactness)**.** Two editions are perceptually equivalent if and only if they have the same class.

This is a modeling assumption and a strong one. The "if" direction follows from the definition of a class. The "only if" direction asserts that a difference of class in any single slot is perceptible, which is an empirical claim about viewers and not a consequence of

the formalism. The earlier volume shows that thresholded similarity is not transitive, and Section 1.7 returns to what that does to the count.

Definition 1.6 (Rules)**.** A *rule set* $\mathcal{R}$ for a production consists of a subset $U_j \subseteq T_j^p$ of *admissible takes* at each slot and a relation $C_j \subseteq T_j^p \times T_{j+1}^p$ of *compatible neighbors* for each $j < m$, both depending only on the classes of the takes involved. The set of *admissible editions* is

$$\mathrm{Ed}(p, \mathcal{R}) = \big\{(\tau_1, \dots, \tau_m) : \tau_j \in U_j \text{ for all } j,\ (\tau_j, \tau_{j+1}) \in C_j \text{ for all } j < m\big\}.$$

Two rules matter here. The *direction rule* D asks that consecutive takes agree in screen direction unless one is neutral, so that $(\tau, \tau') \in C_j$ if and only if $d_\tau = d_{\tau'}$ or $N \in \{d_\tau, d_{\tau'}\}$. This is the familiar convention that a cut must not reverse the direction in which an actor or machine appears to travel [2]. The *agreement rule* G asks that every take agree with the required appearance, so that $U_j = \{\tau : a_\tau = r_j\}$, the set of takes that are not stale. The production begins with the rule set $\mathcal{R}_0 = \{D\}$. Nobody has written the agreement rule down, and the first viewer report will cause it to be written. The amended rule set is $\mathcal{R}_1 = \{D, G\}$.

Proposition 1.7 (Counting editions)**.** *Fix a rule set. Let K_j be the set of classes at slot j whose takes are admissible, let $w_j(k)$ be the number of takes in class $k \in K_j$, and let $A_j \in \{0,1\}^{K_j \times K_{j+1}}$ be the matrix with entry* 1 *if the pair of classes is compatible and* 0 *otherwise. The number of admissible class tuples is*

$$N^{\mathrm{perc}} = \mathbf{1}^\top A_1 A_2 \cdots A_{m-1} \mathbf{1},$$

and the number of admissible take tuples is

$$N^{\mathrm{take}} = w_1^\top A_1 \ \mathrm{diag}(w_2)\, A_2 \ \mathrm{diag}(w_3) \cdots \mathrm{diag}(w_{m-1})\, A_{m-1}\, w_m.$$

If the rendering and export map that assembles a take tuple into a file separates distinct tuples at the byte level, then N^{take} is also the number of byte-distinct admissible editions. Without that assumption it is only the number of tuples.

Proof. The (k, k') entry of $A_1 \cdots A_j$ is the number of compatible class sequences from $k \in K_1$ to $k' \in K_{j+1}$, by induction on j: the entry of the product $A_1 \cdots A_{j-1} A_j$ at (k, k') is $\sum_{k''} (A_1 \cdots A_{j-1})(k, k'') A_j(k'', k')$, which counts sequences by their penultimate class. Summing all entries gives N^{perc}. In the weighted case each class k at slot j contributes the factor $w_j(k)$, which the diagonal matrices and the end vectors supply. □

This is the standard transfer-matrix count for paths in a layered graph [3]. It also gives ranking and unranking for free. The number of completions from each class at each slot is the vector $A_j \cdots A_{m-1}\mathbf{1}$, and choosing classes slot by slot in a fixed order while subtracting these counts maps an integer in $[0, N^{\mathrm{perc}})$ to a class tuple and back.

Corollary 1.8 (Sustained audiences from independent choices)**.** *If $|K_1| \geq q$ and every class in K_j is compatible with at least κ classes in K_{j+1} for each $j < m$, then $N^{\mathrm{perc}} \geq q\kappa^{m-1}$. In particular a*

production with a single admissible opening class and two compatible choices at each of thirty later slots supports $2^{30} = 1{,}073{,}741{,}824$ *perceptually distinct editions, and thirty-three such slots support* $2^{33} = 8{,}589{,}934{,}592$.

Proof. Every partial path through slot j extends in at least κ ways to slot $j+1$, so the number of paths multiplies by at least κ at each of the $m-1$ transitions, starting from at least q. □

The corollary is a lower bound for a hypothetical production and not a measurement of any actual one. It shows that the number of independent, perceptible, compatible choices required to give a world audience distinct editions is small. The count that matters is not the number of bits in a film but the number of such choices, and the hypothesis that they are available at every slot is where the practical difficulty lies.

1.5 Three Local Productions

Three productions film the scene of Table 1.2. They differ in what they happened to shoot, and the differences are chosen to show three different behaviors. The pools are listed in Table 1.3. Each entry gives the number of perceptual classes with a given direction, and the footage in each class exists in several distinct takes: three for each class in production A, two in production B, and forty in production C.

Slot	Production A (3 takes/class)	Production B (2 takes/class)	Production C (40 takes/class)
1	ok: $L^2R^2N^1$	ok: L^2R^3	ok: L^1R^1
2	ok: $L^2R^1N^1$; stale: L^1	ok: L^2R^2; stale: R^2	ok: L^1; stale: L^1
3	ok: L^1R^2; stale: R^1	ok: L^3R^2; stale: L^2	ok: L^1R^1
4	ok: N^2L^1; stale: R^1	ok: L^2R^2	ok: L^1R^1
5	ok: $L^1R^2N^1$; stale: L^1	ok: L^2R^2; stale: R^3	ok: L^1; stale: R^1

Table 1.3: Classes available to each production. A superscript is the number of perceptual classes with the given screen direction (L, R, or neutral N). A class is stale when its takes show some strip in frame in the wrong appearance. The numbers are the parameters of an illustrative model.

Production A shot several neutral takes, which interrupt the chain of screen directions. Production B shot no neutral takes, so every slot is committed to a direction. Production C shot very little but encoded each class in forty takes, for instance by re-exporting or by regenerating grain and compression. The counts by Proposition 1.7 are in Table 1.4, and they are reproduced by the script that accompanies the chapter.

The table shows the central difference between the two kinds of count. Production C has more than two hundred million take tuples and two perceptual editions. If the export map separates take tuples at the byte level, a promise that no two viewers receive the same file is trivially kept by C, while a promise that no two viewers receive perceptibly the same film cannot be kept by C for three viewers. The ratio of the two counts for A is about 243, and for

	Perceptual editions		Take tuples	
Production	under $\mathcal{R}_0$	under $\mathcal{R}_1$	under $\mathcal{R}_0$	under $\mathcal{R}_1$
A	494	238	120,042	57,834
B	320	96	10,240	3,072
C	2	1	204,800,000	102,400,000

Table 1.4: Admissible editions before and after the agreement rule is added.

B it is 32. The tuple count measures the storage of the production and the perceptual count measures what the audience can experience. The amendment from $\mathcal{R}_0$ to $\mathcal{R}_1$ halves both counts for C, and the second halving takes C from two perceptual editions to one.

1.6 A Reported Error and Its Footprint

1.6.1 Reports and verdicts

A viewer's report is a claim about what the viewer saw, and it can be mistaken in two different ways. The viewer may misdescribe the footage, or the footage may misdepict the story. The first is settled by the footage the viewer was actually assigned, and the second by the script history. A verdict therefore needs both, and the ledger of Section 1.8 is what connects the report to the footage.

Definition 1.9 (Report and verdict)**.** A *report* is a tuple (u, f, δ, j, s, x) stating that viewer u, viewing film f on date δ, saw strip $s \in V_j$ in appearance x at slot j. Its *verdict* is determined in three steps. First, the edition $e = \mathrm{Ed}_{u,f}(\delta)$ that the ledger records for u on date δ is resolved, and τ_j is its take at slot j. Second, the *verified appearance* $a_{\tau_j}(s)$ is compared with the claim. The report is *unsupported* if $a_{\tau_j}(s) \neq x$, and no further step is taken. Third, if the claim matches the take, the verified appearance is compared with the required appearance $r_j(s)$ of Definition 1.3. The report is *upheld* if $a_{\tau_j}(s) \neq r_j(s)$, which is to say that the take was stale at that slot, and *rejected* otherwise.

The verdict depends on the footage and the history together. A viewer who reports b unmown at slot 3 of the full script, where b must be mown, is mistaken if the assigned take shows b mown, and the report is unsupported. A viewer who reports b mown at the same slot is describing the footage correctly, and the report is rejected, because the script requires it. A viewer whose assigned take showed b unmown has a report that is upheld. A report about a superseded edition is adjudicated in the same way, since the ledger returns the take that was assigned on δ even after a repair has replaced it. Upholding such a report does not by itself call for a new rule, because the defect may already have been removed by an earlier amendment. The verdict then confirms that the earlier edition was defective and ends there.

Adjudicating against the history also explains how a missed strip can be a correct depiction. If the script is $p_a\, p_c$ and the strip b is skipped, the fold $M(h^*_{\le t})$ never contains b, the

required appearance of b is unmown at every slot, and a take that shows b mown is the error. The same take is correct under the full script $p_a\, p_b\, p_c$ at slots 3 and 5. Whether a character skips a strip because the story requires it is fixed by the history that was committed, and a generator that repaired every unmown strip it found would erase the story that depends on it. The distinction between an error in the film and an act by a character is therefore located in the history layer, and deciding it requires the history to say which departures from competence are intended. Section 1.10 lists this as a question for a later chapter.

1.6.2 The footprint of an amendment

The report that triggers the amendment concerns a reversion: a strip is shown unmown after its pass. For each strip s let k_s be the position of p_s in the script, so that $k_a = 1$, $k_b = 2$, $k_c = 3$.

Definition 1.10 (Footprint)**.** The *footprint* of strip s is $F(s) = \{j : s \in V_j\}$, the set of slots that show it. Its *late footprint* is $F^{\text{late}}(s) = \{j \in F(s) : t_j \geq k_s\}$, the slots at which the strip must appear mown.

For the plan of Table 1.2 the footprints are $F(a) = \{1,2,3,5\}$, $F(b) = \{1,3,5\}$, and $F(c) = \{1,4,5\}$, and the late footprints are $F^{\text{late}}(a) = \{2,3,5\}$, $F^{\text{late}}(b) = \{3,5\}$, and $F^{\text{late}}(c) = \{5\}$. A reversion of strip a can occur at slots 2, 3, or 5 and nowhere else. The early part of the footprint is where the opposite error lives, a strip shown mown before its pass. Slot 4 is such a slot for strip c: production A shot a take that shows c already mown at slot 4, and a rule that forbids only reversion would leave that take in place. The agreement rule G removes both kinds of error because it compares the appearance with the state in both directions.

Proposition 1.11 (Effect of the amendment)**.** *Let $\mathcal{R}_1 = \mathcal{R}_0 \cup \{G\}$. An edition of $\mathrm{Ed}(p, \mathcal{R}_0)$ belongs to $\mathrm{Ed}(p, \mathcal{R}_1)$ if and only if none of its takes is stale. Consequently the editions invalidated by the amendment, for every viewer of the production, are exactly those that contain a stale take at some slot, and $\mathrm{Ed}(p, \mathcal{R}_1) \subseteq \mathrm{Ed}(p, \mathcal{R}_0)$.*

Proof. The compatibility relations of the two rule sets are the same, so an edition is admissible under $\mathcal{R}_1$ if and only if it is admissible under $\mathcal{R}_0$ and each take satisfies the admissibility condition added by G, which is that the take is not stale. □

The proposition makes the reach of one report explicit. The amendment is a change to the production, and so it removes the stale takes from every edition that contains one, including those of viewers who never noticed anything wrong. In production A the amendment removes 256 of the 494 perceptual editions, and in production B it removes 224 of 320.

1.6.3 Propagation through neighbors

The agreement rule acts slot by slot and so propagates through the footprint without chaining. The direction rule acts between neighbors, and it is the neighbor relation that makes a local

repair spread. Replacing a stale take at slot j by an admissible take with a different screen direction can break compatibility with the take at slot $j-1$ or $j+1$, which then has to be replaced, which can break its other neighbor in turn. A neutral take stops the chain.

Lemma 1.12 (Containment)**.** *Let $e = (\tau_1, \dots, \tau_m) \in \mathrm{Ed}(p, \mathcal{R})$ and suppose $a < b$ are slots with $d_{\tau_a} = d_{\tau_b} = N$, where $\mathcal{R}$ has the direction rule as its only relation between neighbors. Let e' be any tuple that agrees with e outside the slots $a+1, \dots, b-1$, whose takes are admissible, and whose neighboring pairs are compatible for the slots $a, \dots, b$. Then $e' \in \mathrm{Ed}(p, \mathcal{R})$.*

Proof. The unary conditions hold at slots outside (a, b) because e' agrees with e there, and they hold inside by hypothesis. The neighbor conditions for pairs of slots both outside $[a, b]$ hold because those slots are unchanged. The conditions for pairs inside $[a, b]$ hold by hypothesis. The only remaining pairs are $(a-1, a)$ and $(b, b+1)$, which involve unchanged takes. So all conditions hold. □

The lemma bounds the damage of a repair by the distance to the nearest neutral takes on each side, provided a completion exists inside the interval. It was also checked exhaustively on production A, over all 3,420 replacements of this kind that the pools permit, with no violation. Production A has neutral takes in several slots, and so its repairs stay local. Production B has none, and its repairs can run from one end of the scene to the other. The maximum number of slots that the matched repair of Section 1.7 changes for one viewer is 2 in A and 3 in B, in a scene of five slots.

1.7 When a Repair Collides

1.7.1 The promise and its limit

Suppose a release promises that no two viewers of a production receive the same edition. The promise has two readings. In the first, the assembled files differ. In the second, the viewers can tell the films apart. The first is cheap to keep when the export map separates take tuples. A keyed hash of a release identifier and a private viewer identifier gives each viewer a stable seed [5], and for 10^9 viewers and 64-bit keys the expected number of colliding pairs is $10^{18}/2^{65} \approx 0.027$. Collisions of seeds can also be removed entirely by assigning each viewer a distinct integer and passing it through a keyed permutation of $[0, N)$, with the domain adjusted by cycle walking [6]. Composing with the unranking that Proposition 1.7 provides then gives each viewer a distinct class tuple, so long as the audience is no larger than N^{perc}.

The second reading is the one that matters for the audience. It is a promise about equivalence classes, and the following statement says when it can be kept within one production.

Proposition 1.13 (Feasibility of distinct editions within a production)**.** *Let d be a pseudometric on the editions of a production p such that editions of the same class are at dissimilarity 0, and let*

Assumption 1.5 hold. An audience of U viewers of p can be given pairwise inequivalent editions if and only if $U \leq N^{\text{perc}}$. If the promise is strengthened to require that every pair of editions have dissimilarity greater than a tolerance $\varepsilon \geq 0$, the largest audience that can be served is the size M_ε of a largest subset of editions with pairwise dissimilarity greater than ε, and $M_\varepsilon \leq N^{\text{perc}}$.

Proof. If $U \leq N^{\text{perc}}$, assign distinct classes. If $U > N^{\text{perc}}$, two viewers share a class by the pigeonhole principle, and by the assumption their editions are equivalent. For the second statement, a set of editions with pairwise dissimilarity greater than $\varepsilon \geq 0$ has no two members of the same class, because members of one class are at dissimilarity 0 by hypothesis, and so its size is at most the number of classes. □

The pseudometric hypothesis is what the upper bound uses. Without the requirement that members of one class be at distance zero, a separated set could contain two editions of the same class and the bound by the number of classes would not follow.

The strengthened promise has a smaller bound than the number of classes whenever similarity is thresholded and not exact. Take four class representatives on a line at positions 0, 0.6, 1.2, and 1.8 with dissimilarity equal to distance, and $\varepsilon = 1$. There are four classes, and no three of the positions are pairwise more than 1 apart, so $M_1 = 2$. The example is a toy with assumed dissimilarities, and it stands for the statement of the earlier volume that thresholded similarity is not transitive. For the present chapter the consequence is that a sustainable audience is a packing number, and that it is bounded above, but only bounded, by the count of Proposition 1.7.

1.7.2 Local and worldwide uniqueness

The counts of Proposition 1.7 and the statement of Proposition 1.13 guarantee distinct editions within each local production. They say nothing about two viewers in different productions. A worldwide promise needs a comparison of editions across productions, and the comparison is not supplied by anything defined so far.

Let p range over the local productions of a release, write Ed_p for the admissible editions of p under its current rules and footage, and let d be a pseudometric on the disjoint union $\bigsqcup_p \text{Ed}_p$. Two editions are *equivalent* when $d = 0$, and the equivalence classes are the *global classes*. Write G for the set of global classes and $G_p \subseteq G$ for those that some edition of p realizes. Editions of different productions can be equivalent. This happens at passages in which no local actors appear, for instance an establishing view of the lawn that two productions have shot from the same position under the same conditions, or footage that the productions share. It does not happen at passages in which the cast, the language, or the setting differ perceptibly. A sufficient condition for the global classes of different productions to be disjoint is therefore that every edition of every production contains at least one slot at which the production's own cast appears and differs perceptibly from every take of every other production at that slot. Whether this holds is a fact about the footage and the viewers, and it has to be established. The scene of the running example does not model a cast, and the counts of Table 1.4 are counts within one production.

Proposition 1.14 (Worldwide feasibility)**.** *Let U_p be the number of viewers in locale p, and suppose each viewer must receive an edition of the production of the viewer's own locale. Pairwise inequivalent editions for all viewers exist if and only if for every set P of productions*

$$\Big|\bigcup_{p\in P} G_p\Big| \;\geq\; \sum_{p\in P} U_p.$$

If the sets G_p are pairwise disjoint, the condition reduces to $U_p \leq |G_p|$ for each p separately, and the allocation can be carried out locale by locale.

Proof. A viewer of locale p must be assigned a global class in G_p, and distinct viewers must receive distinct global classes. By Hall's theorem [4] this is possible if and only if every set X of viewers satisfies $|\bigcup_{u\in X} G_{p(u)}| \geq |X|$, where $p(u)$ is the locale of u. The union depends only on the set P of locales that X meets, and $|X|$ is largest when X contains every viewer of those locales, namely $\sum_{p\in P} U_p$. So the condition is equivalent to the stated inequality for all P. If the sets are pairwise disjoint, $|\bigcup_{p\in P} G_p| = \sum_{p\in P} |G_p|$, and the inequality for all P is equivalent to the inequality for singletons, since the singleton case implies every sum. □

If the promise is strengthened to tolerance $\varepsilon > 0$, the same reading applies with M_ε taken over the union. Locale-by-locale allocation is then sound when editions of different productions are more than ε apart, which is a stronger requirement than disjointness of classes. The chapter's allocation results should accordingly be read as guarantees within a production, and as worldwide guarantees only under the separation condition that Proposition 1.14 identifies.

1.7.3 Repairs and their collisions

A correction forces some viewers to receive a new edition, and the new edition can coincide with another viewer's. Let I be the set of viewers whose edition is invalidated by Proposition 1.11, let S be the remaining viewers, whom this subsection calls the survivors, and let F be the set of admissible classes not held by the survivors. The model of this subsection holds the survivors' editions fixed. A repair is an injective map $f : I \to F$ with $f(u)$ lying in a set $C_u \subseteq F$ of acceptable replacements, for instance the classes within a given Hamming distance of the viewer's previous class.

Proposition 1.15 (Feasibility of collision-free repair with survivors held fixed)**.** *An injective choice $f(u) \in C_u$ for all $u \in I$ exists if and only if for every subset $X \subseteq I$ the union $\bigcup_{u\in X} C_u$ has at least $|X|$ elements.*

Proof. This is Hall's theorem on systems of distinct representatives for the family $(C_u)_{u\in I}$ [4]. □

The condition fails more easily than it appears, and a sequential allocator can fail even when the condition holds. If two invalid viewers have $C_{u_1} = \{x, y\}$ and $C_{u_2} = \{x\}$, an allocator that serves u_1 first and tentatively gives u_1 the class x leaves u_2 with nothing,

although $u_1 \mapsto y$ and $u_2 \mapsto x$ is a valid repair. The example shows only that a tentative assignment among the viewers being repaired must be open to reconsideration. All of these viewers receive a new edition in any case, so reconsidering the tentative assignment disturbs no one.

Holding the survivors fixed is itself a modeling choice, and it can be the obstruction. Suppose an invalid viewer u has acceptable replacements $\{x\}$, a survivor v holds x, and the class y is admissible, unheld, and acceptable for v. Then $C_u \cap F = \varnothing$ and the model reports failure, although moving v to y would give a valid repair. Allowing survivors to move is a broader optimization. A natural form assigns to each viewer u a cost $c(u, e')$ for receiving the class e', with $c = 0$ when e' is the viewer's current valid edition and $c = \lambda + h$ otherwise, where h is the number of slots in which e' differs from the viewer's previous edition and $\lambda \geq 0$ is the cost of disturbing a viewer at all. The problem is then a minimum-cost injective assignment of all viewers to admissible classes. It is feasible if and only if Hall's condition holds for the family of all viewers' acceptable sets, where each survivor's set contains the survivor's current class. That condition is weaker than the condition of Proposition 1.15, and it makes the price of disturbing a valid edition an explicit parameter. The experiments below use the narrower model and do not measure that price.

The model is run on audiences drawn at random from the editions admissible under $\mathcal{R}_0$, as a stand-in for the keyed permutation. Production A has 60 viewers and production B has 30. After the amendment the viewers whose editions are invalid are 31 of 60 in A and 20 of 30 in B. A repair that gives each invalid viewer the nearest admissible edition, with ties broken in a fixed order and no check for collisions, leaves 14 viewers of A and 11 viewers of B sharing an edition with someone else. A repair that processes the invalid viewers in turn and gives each the nearest edition not yet held changes 37 slots in A, the same number as the unconstrained repair, and 27 in B, one more than the unconstrained 26. A repair that minimizes the total number of slots changed over all injective maps from I into F also changes 37 slots in A and 27 in B. In these instances the price of removing collisions is zero in A and one slot in B, and greedy allocation did not fail. Production C is the case that does not admit a repair at all. Its audience of two viewers is served by the two editions of $\mathcal{R}_0$ and cannot be served by the single edition of $\mathcal{R}_1$. No assignment exists, with survivors fixed or free, because $U = 2 > 1 = N^{\mathrm{perc}}$ and Proposition 1.13 applies. The correction cannot take effect without either breaking the promise or adding footage.

1.7.4 Capacity under correction

The result for production C states the question that the book carries forward. Say that a production has, at a date s, a *capacity* $\mathrm{Cap}(p, s) = N^{\mathrm{perc}}$ computed from the rules and the footage in force at s. By Proposition 1.13 a promise of distinct editions to U_p viewers is keepable at s if and only if $U_p \leq \mathrm{Cap}(p, s)$. A production is open to correction when upheld reports can add rules and replacement footage can be added. The question is how much variation a production can sustain while remaining open to correction. It has a precise form.

Definition 1.16 (Robustness)**.** Let $\mathfrak{A}$ be a family of rule sets that extend the current rule set $\mathcal{R}$. A production p with audience U_p is *robust to* $\mathfrak{A}$ if $N^{\text{perc}}(p, \mathcal{R}') \geq U_p$ for every $\mathcal{R}' \in \mathfrak{A}$.

Production C at $U = 2$ is not robust to $\{\mathcal{R}_1\}$: it has zero spare capacity at the premiere, with $\text{Cap} = 2 = U$, and the one correction that the first report triggers takes the capacity to 1. Production A with 60 viewers is robust to $\{\mathcal{R}_1\}$, since $238 \geq 60$, and so is B with 30, since $96 \geq 30$. Robustness to a given family is a lower bound on the footage a production must hold before the first report, and the families of amendments that can be anticipated, and the capacity they cost, are the subject of a later chapter.

1.8 Editions Across Time

An edition is not fixed once assigned. It changes when a correction arrives, and a viewer may want to see the edition that was available on an earlier date. The earlier volume separates history, state, and appearance, and the same separation applies one level up. Distinguish the *archive* of a production, which holds every take it has ever supplied, from the *admissible pool* at a date, the takes that the rules in force allow. A take that is removed from admissibility is not removed from the archive.

Definition 1.17 (Ledger)**.** A *ledger* for a film f is a finite sequence of dated events of two kinds. A *premiere* event (t, f, u, e) assigns the edition e, a tuple of take identifiers, to viewer u. A *correction* event $(t, f, \mathcal{R}', \Delta T, \{(u, e'_u)\}_{u \in I})$ takes effect at the single time t and does three things together. It replaces the rule set by $\mathcal{R}'$, it adds the footage ΔT to the archive and the admissible pool, and it replaces the edition of every viewer $u \in I$ whose edition is no longer admissible by e'_u. The *edition of f available to u at date δ*, written $\text{Ed}_{u,f}(\delta)$, is the edition named by the last premiere or correction event for u with time at most δ, and is undefined before the premiere.

The edition is a right-continuous step function of δ, and a viewer who moves the date back before a correction recovers the edition that the correction replaced. The ledger also serves the adjudication of Definition 1.9, since it returns the take a viewer was assigned on the date of viewing even when a later correction has replaced it.

The amendment and its repairs are recorded as one event on purpose. Let $\mathcal{R}(\delta)$ and $T(\delta)$ be the rules and the admissible pool in force at δ.

Proposition 1.18 (Admissibility invariant)**.** *If every correction event repairs every viewer whose edition it invalidates, then for every date δ and every viewer u with a defined edition,* $\text{Ed}_{u,f}(\delta) \in \text{Ed}(p, \mathcal{R}(\delta), T(\delta))$. *If an amendment takes effect at a time t and its repairs are recorded at a later time $t' > t$, then for dates $\delta \in [t, t')$ the slider can show an edition that the rules in force have already ruled inadmissible.*

Proof. Premiere events assign admissible editions, and each correction event leaves every viewer with an edition admissible under the rules and pool that the event installs. Between events nothing changes, so the invariant holds at every date. If the amendment is recorded

alone at t, a viewer whose edition contains a stale take keeps it until the repair is recorded, and for $\delta \in [t, t')$ the rules in force exclude that edition while the ledger still returns it. □

A correction event can fail to exist. By the result for production C, if $U > N^{\mathrm{perc}}(p, \mathcal{R}', T \cup \Delta T)$ no set of repairs satisfies the promise, so the event cannot be formed without enough added footage in ΔT.

Proposition 1.19 (The ledger is not a function of the seeds)**.** *The edition* $\mathrm{Ed}_{u,f}(\delta)$ *is in general not determined by the viewer's seed, the viewer's earlier edition, and the rule set in force at* δ*.*

Proof. Suppose the admissible classes after an amendment are $\{x, y, z\}$ and two invalid viewers u_1, u_2 have x as their nearest admissible class, with ties broken in favor of x, and y is their next nearest. If the audience consists of u_1 and u_2 and they are served in this order, the repair gives u_1 the class x and u_2 the class y. If the audience also contains a survivor u_0 who already holds x, the repair gives u_1 the class y and u_2 the class z. Thus u_1 ends with x in the first case and y in the second, although the seed, the earlier edition, and the rule set of u_1 are the same in both. □

The proposition says that the seed and the rules do not determine the edition, so they cannot regenerate it. The recorded assignment can name the old edition, but showing it again requires more than the name. It requires the exact source assets that the recorded take identifiers reference, including takes that later rules removed from admissibility, and the assembly information, such as cut points, grading, mix, and export parameters, needed to produce the file from the tuple. If an asset or the assembly information has been discarded, the ledger still records which edition the viewer was assigned and cannot show it. In the vocabulary of the earlier volume the ledger is a history, the edition at a date is the state obtained by folding its prefix, and what the viewer sees is an appearance derived from that state, which is regenerable only so long as what it is derived from is kept.

1.9 The Lifetime Repertoire

1.9.1 A premiere, a slider, and a viewing budget

A *release* is a pair (f, t_f) of a film and a premiere instant common to all viewers everywhere. Each local production supplies the footage for its locale, and each viewer in a locale receives an edition from that production's admissible set. Each viewer u has a finite *repertoire* F_u of films, selected and scheduled for u from the enormous space of possible films, and the space itself never has to be enumerated. A *release slider* at date s shows

$$\mathrm{Rep}_u(s) = \big\{(f, \mathrm{Ed}_{u,f}(s)) : f \in F_u,\ t_f \le s\big\},$$

together with the scheduled premieres $\{(f, t_f) : f \in F_u,\ t_f > s\}$, which have no edition yet. Moving the slider back to a date before a repair shows the edition of that date, by Section 1.8, and moving it to a future date shows only the schedule. The viewer who reported the

reverting lawn finds the edition in which the report was made at a slider position before the repair, and the corrected edition at any position after it.

A viewing budget bounds the size of the repertoire. Let B be the viewer's lifetime budget in hours, let ℓ_f be the length of film f in hours, and let $n_f \geq 1$ be the number of times f is viewed. The repertoire is feasible if $\sum_{f \in F_u} \ell_f n_f \leq B$. An illustrative budget of two hours a week over sixty years of fifty-two weeks is $2 \cdot 52 \cdot 60 = 6{,}240$ hours. Films of ninety minutes watched once fill $6{,}240/1.5 = 4{,}160$ places. If each film is viewed on average $1 + r$ times, the number of distinct films falls to $B/(\ell(1 + r))$, which is 2,080 for $r = 1$ and 1,386 for $r = 2$, rounding down. The budget parameters are free, and the point is that the number of films a person can watch is small next to the number of films that exist, and smaller still when revisits are allowed for.

1.9.2 Corrections inside a repertoire

A correction of the lawn takes effect in the viewer's repertoire at the effective time of the correction event of Definition 1.17, and not at the premiere. The edition seen at the premiere remains recoverable from the ledger, provided the assets and assembly information it references are retained, and the edition seen after the correction is the default for later viewings. The counting question then has a second face. It is no longer how many distinct films can be generated, but how many distinct editions can be kept coherent as corrections accumulate, in each production, for each film in each viewer's repertoire. The three productions above are three answers to this question for a single scene.

Proposition 1.20 (Monotonicity for a fixed pool)**.** *Fix the pools of takes. If rule sets satisfy $\mathcal{R} \subseteq \mathcal{R}'$ in the sense that every constraint of $\mathcal{R}$ is a constraint of $\mathcal{R}'$, then $\mathrm{Ed}(p, \mathcal{R}') \subseteq \mathrm{Ed}(p, \mathcal{R})$, and the number of perceptual classes and the packing number M_ε do not increase.*

Proof. An edition satisfying all constraints of $\mathcal{R}'$ satisfies all those of $\mathcal{R}$, so the admissible sets are nested. A class that has an admissible edition under $\mathcal{R}'$ has one under $\mathcal{R}$, so the number of classes does not increase. A separated subset of $\mathrm{Ed}(p, \mathcal{R}')$ is a separated subset of $\mathrm{Ed}(p, \mathcal{R})$, so M_ε does not increase. □

The proposition establishes one thing: adding constraints to a fixed pool cannot increase capacity. It does not say that capacity falls with every report. An upheld report may expose a violation of a rule already in force, in which case the defect lay in the assembly or in the assigned take and no new rule is added, and a production may add replacement footage as part of a correction, which enlarges the pool and can raise capacity above its earlier value. The ways to restore capacity after a rule has been added are to add footage, to add classes that pass the rules, or to relax a rule in a way that the history permits.

1.9.3 Order and sequence

The budget treats films as interchangeable items. They are not. A later film can change what an earlier one means, in the way that a revelation late in the engine scene reclassifies the

opening repair as diligence, negligence, or concealment. If seeing a film again can change the meaning of an earlier scene, a repertoire that assumes each film is watched once will overestimate the distinct films a person has time for, and the order of release becomes a variable. The same film released in year ten of a viewer's life and released in year one does not do the same work for the repertoire, because what the viewer has seen already differs. This is a question about sequences of films and not about one film, and the present chapter only records it.

1.10 Questions the Example Opens

The example was chosen because it is small, and the questions it raises are not. Each of the following needs a chapter of its own, and the order below is the order in which the example forced them.

The first is the measurement of equivalence. Assumption 1.5 treats perceptual classes as given, and Proposition 1.13 shows that the correct count under thresholded similarity is a packing number that depends on a dissimilarity no one has measured. A chapter on equivalence has to say what is detected, what changes comprehension, and what changes affective response, and how each defines its own dissimilarity, so that the promise of distinct editions can be stated against one of them. It also has to define the comparison across local productions, on which the worldwide promise of Section 1.7.2 depends.

The second is the propagation of corrections through state that crosses slots. In the example the agreement rule acts on single slots through the footprint, and the direction rule acts on neighbors. Continuity in a real film involves relations of both kinds and others that span arbitrary distances, such as a glass held in one hand across a cut. The earlier volume treats agreement on overlaps by sheaf conditions, and the question is how a correction derived from one violated overlap finds all the other overlaps it implicates.

The third is the adjudication of reports, which Definition 1.9 splits into the viewer's claim, the footage the ledger shows the viewer was assigned, and the script history. The distinction between an error in the film and an act of a character is located in the history. A history that records only what happened cannot say which departures from competence the story intends. The history therefore needs annotation by dramatic function, and the corresponding formalism has to guarantee that a correction cannot delete the evidence a later reinterpretation depends on.

The fourth is the allocation of editions under change. Proposition 1.15 states when a collision-free repair exists, and Proposition 1.19 shows that the edition of one viewer depends on the others. The algorithmic question is how to repair with the least disturbance, including the price λ of moving a viewer whose edition was valid, and how to make the result reproducible from a ledger of bounded size and a retained archive. The robustness of a production to families of anticipated amendments (Definition 1.16) belongs here as well.

The fifth is exclusion. Remark 1.2 separates exact subtraction by digest, which is arithmetic, from approximate subtraction, which depends on a representation and a threshold

and interacts with the packing numbers above. A chapter on exclusion has to count the editions that survive a declared policy and has to keep that count apart from any question of legal status.

The sixth is noise overcapture, the expense of distinguishing variations that have no bearing on what can happen, what the camera can show, or what the audience can infer. A wet plate matters if it bears on an action, a later shot, or something visible, and otherwise a generator that tracks it spends capacity on nothing. The earlier volume's quotient of histories by future equivalence is the right tool, and the chapter has to extend it from futures to what is displayed.

The seventh is sequence. The repertoire budget of Section 1.9 treats films as items, and the order of viewing and of release changes what each film means. A chapter on sequence has to say what a repertoire is as a structure on films and not as a set.

The accompanying script `verify_ch01.py` reproduces every count in this chapter, including the transfer-matrix totals checked against exhaustive enumeration, the footprints of Section 1.6, the exhaustive check of Lemma 1.12, and the audience experiments of Section 1.7.

Chapter 2

Perceptual Equivalence and the Measurement of Difference

Chapter 1 counted editions by assuming that perceptual classes exist and that two editions are equivalent exactly when they share a class at every slot. The assumption made the counts exact, and every later chapter has used it. This chapter replaces it with an account of dissimilarity and says what the replacement permits. The account separates three things that a viewer can notice, the presence of a difference, its bearing on what the story means, and its effect on how the film feels, and gives each its own pseudometric. It shows what happens to the counts when the equivalence is replaced by a threshold, where the replacement fails to behave like an equivalence, and how the packing and covering numbers that take its place depend on the question the production wants to answer. It then defines the comparison across productions that the worldwide promise of Chapter 11 will need.

The chapter proves what the formalism permits and marks the places where evidence about viewers has to take over from mathematics. The lawn scene supplies the numbers, and the choices of pseudometric used for them are illustrations. None of the figures below is a measurement of any audience.

2.1 What Exact Equivalence Assumed

Exact equivalence says that a viewer cannot tell two editions apart if and only if they have the same class at every slot. It has two halves. If the classes agree at every slot, the viewer cannot tell the editions apart, which says that the class partition is fine enough to hide nothing a viewer could notice. If some class differs, the viewer can tell them apart, which says that every class boundary is perceptible. Both halves are strong, and the book needs only that the counts be computed from some partition. What the first half guarantees is that the editions counted as distinct are not secretly the same, and what the second half guarantees is that no distinct editions are being merged. A production that wanted to promise that no two viewers get the same film would rely on the second half, and a production that wanted to

promise each viewer something perceptibly different would rely on the first.

A more careful account does not partition editions into classes in the first place. It gives each pair of editions a number measuring how different they are.

2.2 Three Notions of Difference

A *pseudometric* on editions is a function d with $d(x, x) = 0$, symmetry, and the triangle inequality, without requiring that $d(x, y) = 0$ implies $x = y$. The editions at distance zero from one another form an equivalence class, and d is a metric on the classes. A pseudometric therefore carries exact equivalence inside it: the equivalence is the relation $d = 0$.

Three pseudometrics are distinguished by what they measure.

Definition 2.1 (Detection, comprehension, affect)**.** The *detection* dissimilarity d_{det} measures whether a viewer can notice, on comparison, that two editions differ. The *comprehension* dissimilarity d_{comp} measures whether the two editions support different inferences about what happened in the story. The *affective* dissimilarity d_{aff} measures whether they produce different responses of feeling, tension, or attention.

The three can disagree. Two editions that differ only in the framing of a shot are detectably different and have the same comprehension. Two that differ in whether grass reappears have different comprehension, and may or may not feel different. Two that differ in the direction of a long take differ in affect and may not differ in what they say happened.

For the lawn scene the following instances are used, as illustrations. Detection is the Hamming dissimilarity on class tuples, the number of slots at which two editions use different classes. Affect is the Hamming dissimilarity on direction sequences, so that two classes of the same screen direction are affectively alike. Comprehension is the Hamming dissimilarity on the pattern of stale classes, because the only narrative information that the scene carries is the state of the grass, and an edition conveys a different story only where a stale class misstates it. The pattern of stale slots records where the film disagrees with the script, which is not necessarily what story it conveys to a viewer, and the choice is an illustration for that reason. The editions at distance zero under each of these are then counted. Table 2.1 gives the numbers of editions that remain distinguishable.

The table carries a result that is worth stating plainly. Once the agreement rule is in force, every admissible edition of every production tells the same story, so the lawn scene has exactly one edition in the sense of comprehension. All of the variation that the book counts is variation of detection and, to a lesser extent, of affect. This is not a defect of the counting. It is what it means for the rules to be correct: the agreement rule is the requirement that editions not differ in what they say. The variation that remains is variation in how it is said. Production B has two affective classes, all its slots committed to the left or all to the right, and 96 detectable ones.

Production	Rules	Detection	Affect	Comprehension
A	direction rule	494	50	8
A	plus agreement rule	238	40	1
B	direction rule	320	2	5
B	plus agreement rule	96	2	1
C	direction rule	2	1	2
C	plus agreement rule	1	1	1

Table 2.1: Number of classes of editions at distance zero under each of three illustrative pseudometrics, for the lawn scene.

2.3 What an Experiment Would Have to Measure

Each of the three pseudometrics names a quantity that a study could in principle estimate, and none has been estimated here.

For detection, a study would present pairs of editions to viewers and record how often the viewer reports a difference, with the pairs chosen to span the slots and classes of the production. The result is a measure of how detectable each class pair is, and the question for the formalism is whether the resulting dissimilarity is close to a sum over slots, as the Hamming choice assumes. A viewer might detect one large difference among five small ones that it ignores, and then the dissimilarity is not additive. Raw pairwise detection frequencies also need not satisfy the triangle inequality. If viewer data resist a pseudometric model, the formalism can be kept by taking the shortest-path closure of the measured dissimilarities, which is the largest pseudometric below them, or by treating the pairs that viewers can tell apart as a graph and asking for independent sets in it, which loses the metric structure and keeps the packing question.

For comprehension, a study would ask viewers questions about the story after watching one edition, and compare the answers across editions. The measure is whether the distribution of answers differs, and the interesting cases are those in which two editions give different answers to a question that the script history answers one way. These are the cases that the agreement rule exists to prevent.

For affect, a study would collect continuous or retrospective ratings of tension, interest, or emotion during viewing, and compare them across editions. This is the hardest of the three, since viewers differ in how a given shot makes them feel and the response to an edition depends on what the viewer expects of the film.

For each, three numbers have to be fixed: a tolerance below which the difference is treated as none, the population over which the measure is taken, and the stability of the measure across viewers. The formalism of this chapter depends on the tolerance, and the later chapters on the population.

2.4 Thresholds Are Not Equivalences

If the tolerance is ε, two editions can be called *ε-similar* when their dissimilarity is at most ε. For $\varepsilon = 0$ this is the exact equivalence. For $\varepsilon > 0$ it is reflexive and symmetric and it is not transitive.

Proposition 2.2 (Failure of transitivity)**.** *Let d be a pseudometric and $\varepsilon > 0$. The relation $d \leq \varepsilon$ is reflexive and symmetric and need not be transitive: an edition x can be ε-similar to y, and y to z, while x and z are not. The triangle inequality gives only $d(x, z) \leq 2\varepsilon$.*

Proof. The triangle inequality bounds $d(x, z)$ by $d(x, y) + d(y, z) \leq 2\varepsilon$ and no better in general. A line with points at 0, ε and 2ε has $0 \sim \varepsilon$ and $\varepsilon \sim 2\varepsilon$ and $0 \not\sim 2\varepsilon$. □

The lawn scene shows how much of the failure there is. Take the 238 editions of production A after the agreement rule, under Hamming dissimilarity on class tuples. At $\varepsilon = 1$, two editions are similar if they differ at most at one slot. There are 10,040 triples in which one edition is similar to two others, and in 8,228 of them the two others are not similar to each other. At $\varepsilon = 2$ the numbers are 263,740 and 147,328. In both cases the similarity graph is connected, so the transitive closure of similarity is a single class. A production that tried to turn the threshold into an equivalence by closing it would conclude that all 238 editions are one film. For production B, with 96 editions, the numbers at $\varepsilon = 1$ are 1,440 and 1,344, and at $\varepsilon = 2$ they are 18,240 and 8,352, and the closure has two components, the all-left editions and the all-right ones.

The failure of transitivity means that the count of editions cannot be generalized by counting classes of a similarity relation. Two replacements remain, and they ask different questions.

2.5 Packing and Covering

The *packing number* M_ε is the size of a largest set of editions that are pairwise more than ε apart, as in Chapter 1. The *covering number* N_ε is the size of a smallest set of editions such that every edition is within ε of one of them.

The two answer different questions. The packing number is the number of viewers who can each be given an edition that is perceptibly different from every other viewer's, which is the sustainable audience of a promise of distinctness. The covering number is the number of editions that suffice to represent every possible edition within the tolerance, which is the number of distinct films the audience could be shown if the production were content to show each viewer something within ε of the film the viewer's assignment represents. It is the quantity that a compression of the space of editions at tolerance ε needs, and the logarithm $\log_2 N_\varepsilon$ is the number of bits that an index into such a compression takes.

Proposition 2.3 (Packing and covering)**.** *For every pseudometric on a finite set and every integer tolerance ε of a Hamming-type dissimilarity, $N_\varepsilon \leq M_\varepsilon \leq N_{\lfloor \varepsilon/2 \rfloor}$. For a general pseudometric the last term is $N_{\varepsilon/2}$.*

Proof. A largest set of editions that are pairwise more than ε apart is maximal, so every other edition is within ε of it, and the set covers at radius ε, which gives $N_\varepsilon \leq M_\varepsilon$. For the second inequality, suppose a set of editions is pairwise more than ε apart, and cover the space by balls of radius $\lfloor \varepsilon/2 \rfloor$. Two members of the set in one ball would be within twice that radius of each other, at most ε, a contradiction. So each ball holds at most one member of the set, and there are at least as many balls as members. □

Table 2.2 gives the numbers for the lawn editions after the agreement rule, under Hamming dissimilarity on class tuples. The packing numbers were computed exactly by integer programming. The covering numbers were computed by integer programming within a time limit. They are exact where the bounds meet, and for production A at $\varepsilon = 1$ the solver proved only that the number lies between 25 and 31.

Production	ε	Packing M_ε	Covering N_ε	$\log_2 M_\varepsilon$	$\log_2 N_\varepsilon$
A	0	238	238	7.9	7.9
A	1	54	25 to 31	5.8	4.6 to 5.0
A	2	16	8	4.0	3.0
B	0	96	96	6.6	6.6
B	1	32	16	5.0	4.0
B	2	12	6	3.6	2.6

Table 2.2: Packing and covering numbers of the admissible editions after the agreement rule, under Hamming dissimilarity on class tuples.

The numbers show how sharply the audience that a promise can sustain depends on the tolerance. Production A can serve 238 viewers with exactly distinct editions and 54 with editions that differ at two or more slots, and 16 with editions that differ at three or more. The covering numbers say how many reference editions would stand in for the whole space: eight at $\varepsilon = 2$ for A. This is the same arithmetic that Chapter 1 used for its rate, with a different numerator. The rate of that chapter counts exactly distinguishable editions. The covering count is the quantity for a production that is content with distinguishability up to a tolerance.

The two numbers are not interchangeable, and the choice between them is part of what a production promises. A promise of pairwise distinctness is a packing promise and its audience is bounded by M_ε. A promise that every viewer's edition is within ε of an agreed reference is a covering promise and costs N_ε references. A production that confuses them will state a sustainable audience of N_ε when the audience it can sustain is M_ε, and in Table 2.2 the two differ by a factor of about two.

2.6 Across Productions

The worldwide promise of Chapter 1 compares editions from different productions, and the pseudometric has to be extended to editions that were not assembled from the same pool. Two cases arise.

When the productions share no footage, so that no class at any slot of one is a class at the same slot of the other, every pair of editions from different productions differs at every slot. Under Hamming dissimilarity the cross distance is the number of slots, here 5, and the comparison is trivial for every tolerance below 5.

Proposition 2.4 (Disjoint footage composes)**.** *If two productions share no class at any slot, then for every $\varepsilon < m$ the packing number of the union of their editions is the sum of the packing numbers, and exact inequivalence composes: the number of exactly distinct editions of the union is the sum of the two.*

Proof. Every cross pair is at distance $m > \varepsilon$, so a separated set in each production remains separated when combined, and any separated set of the union splits into separated sets of the two. The same holds with $\varepsilon = 0$. □

When productions share footage the cross distance can be small, and zero. To see the effect, suppose that classes with the same direction, status, and index within their group at a slot are the same footage in productions A and B. Then 4 of the class tuples of the two productions after the agreement rule are identical. The exact count of the union is 330, and not the sum 334, and the effect appears at every tolerance. At $\varepsilon = 1$ the packing numbers are 54 and 32 for A and B and the packing number of the union is 85 and not 86. At $\varepsilon = 2$ the numbers are 16, 12 and 27 against 28. A pair of viewers in two locales who receive editions that share all their footage cannot be given editions that differ.

The calculation is small, but its form is general. Shared footage, such as an establishing shot used in every locale, makes the global audience smaller than the sum of the local audiences, and a production that shares more footage saves storage and loses distinctness. This is the rate trade-off of Chapter 1 at the scale of a release.

2.7 What the Formalism Permits

The formalism permits the following. Given a pseudometric, the equivalence it defines is exact and the counts of Chapter 1 follow. Given a tolerance, the packing number bounds the audience of a distinctness promise and the covering number gives the size of a compression, with the inequality of Proposition 2.3 relating them. Given two productions, the combined audience is the sum when they share no footage and is smaller when they do. All of these hold for any pseudometric, and none of them needs a theory of what viewers perceive.

The formalism does not say which pseudometric describes a given audience, what tolerance is perceptible, or whether the three notions of difference can be treated separately. It does not say whether detection is additive over slots, or whether viewers respond to

the same differences the pseudometric counts. These are questions about people, and the answers have to come from studies that have not been done. Until they are done, the numbers in this chapter show how the quantities behave and not what they are. The remaining chapters take the Hamming dissimilarity on class tuples as a stand-in whenever a tolerance is needed.

The scripts `lawnmodel.py` and `verify_ch02.py` reproduce each figure of this chapter.

Chapter 3

Histories, Folds, and What a Scene Commits To

The verdict of Chapter 1 ends in a comparison with the script history. Chapter 7 relied on that comparison to separate an error in the film from an act by a character: a loose bolt that appears in a shot is an error if the story says the bolt was tightened, and it is a depiction if the story says that the mechanic skipped it. The history layer decides, and the history as the earlier volume defines it records only what happened. This chapter extends it in two ways. It annotates histories with dramatic function, so that a departure from competence can be marked as intended. It also states a guarantee that a correction cannot delete the evidence on which a later reinterpretation depends. A last section extends the quotient of histories by future equivalence from what can happen to what can be shown, so that a generator tracks a detail such as a wet plate only when an action, a later shot, or an inference refers to it.

The governing example is an engine scene in which an ordinary repair is later understood as diligence, negligence, concealment, or disclosure. It is small enough to count exactly, and it is built so that the same footage supports several readings until a later shot decides among them.

3.1 The Engine Scene

A mechanic works on an engine with three bolts. For each bolt the mechanic either tightens it or skips it. If any bolt is skipped, the mechanic either does not notice, or notices and then reports honestly, or notices and reports falsely that all is well. These choices are the events of the story, and a *history* is a complete record of them. There are eight patterns of tightening and skipping. The pattern in which all three bolts are tightened gives one history, since there is nothing to notice or report. Each of the seven patterns with a skipped bolt gives three histories, namely not noticed, noticed and honestly reported, and noticed and falsely reported. The story world therefore has $1 + 7 \cdot 3 = 22$ histories.

The histories fall into four *readings*, which are the ways a viewer who sees the whole

story would describe the repair.

Reading	Histories	What happened
Diligence	1	all bolts tightened
Negligence	7	some bolt skipped, not noticed
Disclosure	7	some bolt skipped, noticed, honestly reported
Concealment	7	some bolt skipped, noticed, falsely reported

Table 3.1: The twenty-two histories of the engine scene and the four readings that partition them.

The script history h^* is the one the production intends: the first two bolts are tightened, the third skipped, and the mechanic notices and reports falsely. It is a concealment.

3.2 Annotating Histories with Dramatic Function

The history h^* departs from competence. A competent mechanic tightens all three bolts, and the competent history is the first of Table 3.1. The departure is an act of the character, and it is also something the film has to be allowed to show. If the history layer records only what happened, a film that shows the third bolt loose depicts something that differs from the competent baseline, and a report that this is a continuity error cannot be answered by the record.

An *annotation* assigns each event of a history a dramatic function. The functions used here are *competent*, for an action that follows the baseline, and *lapse*, *noticing* and *concealing* for departures that the story intends. In h^* the skipping of the third bolt is a lapse, the notice is a noticing, and the false report is a concealing. The annotation does not change which events happened. It changes which departures from the baseline are errors in the film and which are acts of its characters.

Definition 3.1 (Verdict with annotation)**.** The showing is first compared with the script history. If the event it displays differs from the event of the script history, it is an *error*, whatever the annotation says, since an intended lapse can still be misdepicted. If the event it displays is the event of the script history and that event departs from the baseline, the annotation decides: the showing is a *depiction of an act* when the annotation marks the departure as one the story intends. A report that a depiction of an act is an error is rejected.

In the engine scene, a report that the third bolt is shown loose in the close shot is rejected, since the displayed state matches h^*. The same report is upheld as an error if the script history is the all-tightened history and no annotation exists. The shot is the same in both cases and the history differs.

The cost of getting this wrong is counted in editions. The scene as shot has 126 class tuples, from two classes at the opening, seven at the bolt close-up, and three each at the glance and the handoff. Of these, 32 are consistent with h^*, in the sense that no shot in them

contradicts it. If the close shot of the loose third bolt is removed as an error, the number of editions consistent with h^* falls from 32 to 24, and it falls by editions that were telling the story correctly.

3.3 What a Scene Commits To

Each shot class shows something, and what it shows rules out some histories. A class that cuts to the wide view of the bench rules out nothing. A close-up of the first bolt, tightened, rules out every history in which that bolt was skipped. The *commitment* of a class is the set of histories consistent with what it shows, and the commitment of an edition is the intersection of the commitments of its classes, which is the set of histories that the edition leaves open. This is the fold of the earlier volume taken from histories to what can be shown: the edition's commitment is the fiber of the fold over the edition.

The relevant properties of an edition are properties of its commitment $C(e)$.

Definition 3.2 (Properties of an edition)**.** An edition e is *consistent* if $C(e)$ is nonempty and *faithful* if $h^* \in C(e)$. It is *open* to a reading if $C(e)$ contains a history of that reading. It is *open* if it is open to all four readings. It is *recoverable as* a reading if $C(e)$ is nonempty and lies wholly within that reading, so that a viewer who has seen the edition can say which reading holds.

For the engine scene 112 of the 126 editions are consistent, and 32 are faithful. A faithful edition can be recoverable only as the reading that h^* belongs to, which is concealment. Eight of the 32 are recoverable as concealment, and six are open to all four readings. No edition is both. An edition that is open to diligence, whose commitment contains the history with every bolt tightened, cannot contain the glance in which the mechanic notices, and an edition that supplies the glance cannot be open to diligence. A scene therefore cannot both leave every reading available and make one of them recoverable. The production's choice, edition by edition, is which of the two to give the viewer, and a production that wants both ambiguity and a later resolution has to place them in different shots, with the evidence in the later one.

3.4 A Correction Cannot Delete the Evidence

The engine scene is built so that the reading of concealment depends on two classes: the glance in which the mechanic notices, and the handoff in which the report is made without comment. Every edition that is recoverable as concealment uses both. The eight recoverable editions differ only in which of the four close-up classes consistent with h^* they use and in which class of the opening shot, and in neither are they constrained by the evidence.

Proposition 3.3 (Preservation of evidence)**.** *Let E be the set of editions that are recoverable as a reading R, and let an amendment remove a set of classes. The editions recoverable as R after the*

amendment are those of E that use none of the removed classes. In particular, recoverability survives the amendment if and only if some edition of E avoids every removed class, and an amendment that removes a class used by every edition of E leaves none.

Proof. An edition's commitment depends only on its classes, so an edition that avoids the removed classes keeps its commitment and its recoverability, and one that uses a removed class is no longer an edition of the production. □

The numbers follow. Withdrawing the glance in which the mechanic notices leaves none of the eight editions recoverable as concealment, and so does withdrawing the clean handoff. Withdrawing one class of the opening shot leaves four of the eight, since the opening is irrelevant to the evidence. The withdrawal of the glance also halves the number of faithful editions, from 32 to 16. The glance can be withdrawn for reasons that have nothing to do with the story, for example because the rights to the shot are lost or the take is damaged, and when it is, the later reinterpretation depends on a class that no longer exists.

The guarantee that the production should give is a condition on amendments. A correction answering a report is *evidence-preserving* for a reading if at least one edition recoverable as that reading avoids every class the correction removes. The test is on the removed set as a whole. It is not enough that no single removed class is used by every recoverable edition: if two recoverable editions use alternative classes a and b at one slot, neither class occurs in every recoverable edition, and removing both destroys recoverability. For an amendment that is the agreement rule with the script history, the condition holds for the reading to which h^* belongs whenever some faithful edition is recoverable as it, by the proposition below.

Proposition 3.4 (Agreement with the script preserves faithful editions)**.** *An amendment that removes exactly the classes whose commitments exclude h^* leaves every faithful edition in place, and therefore preserves recoverability of the reading to which h^* belongs.*

Proof. A faithful edition has h^* in its commitment, so none of its classes excludes h^*. The amendment removes only classes that exclude h^*, so every faithful edition survives. □

The classes that contradict h^* in the engine scene are five: two classes at the bolt close-up that show the first or second bolt loose, the class that shows the third bolt tight, the glance in which the mechanic does not look, and the honest handoff. Agreement with the script removes those and nothing else. The danger is an amendment adopted on a mistaken verdict, as in the example of the loose third bolt, which removes a class that is consistent with h^* and thereby removes editions that carried the story. Chapter 7's verdict, with the annotation of this chapter, is what prevents it.

3.5 Noise Overcapture

A scene has details that matter and details that do not. The engine scene is shot in a workshop with four plates on a rack, each wet or dry, and nothing in the story refers to them.

A generator that tracked every detail of the shot would carry each plate's state as part of the history, and the number of histories would grow with the details.

The earlier volume handles this by a quotient of histories by future equivalence: two prefixes of a history are identified if they allow the same futures. This chapter extends it from what can happen to what can be shown. Two partial histories are *equivalent for the future* if every later shot, action, and inference that refers to the history treats them alike. A plate matters in exactly three ways: an action depends on it, such as the mechanic slipping on a wet plate, a later shot shows it, or an inference uses it, such as a viewer deciding that the mechanic was washing up and left a bolt for later. Unless one of these refers to the plate, its state can be dropped from the history, and the generator is free to render it any way.

Plates referred to by the future	Classes of histories
0	22
1	44
2	88
3	176
4 (all tracked)	352

Table 3.2: Classes of histories of the engine scene, with four plates each wet or dry, as the number of plates referred to by a later action, shot, or inference grows.

The count doubles with each plate that a later shot refers to, and reaches $22 \cdot 2^4 = 352$ only if all four are tracked. A generator that tracks what the future refers to carries 22 classes if nothing refers to the plates. It carries 44 if a later shot shows one of them wet, and the state of that plate then enters every history from the moment the plate's state is fixed. This is the same distinction between what must be remembered and what need not be that Chapter 4 drew with carry width, and here it applies to the history layer.

3.6 What the Chapter Gives the Others

Chapter 7's verdict uses the annotated history: the third step of the verdict compares the verified appearance with the script history and consults its annotation to decide whether a departure is an error or an act. Chapter 8's footprint and correction event take their evidence requirement from Proposition 3.3: a correction is formable only if it is evidence-preserving for the readings the production has promised to keep available. Chapter 14, on sequels and reinterpretation, begins from the property that a scene can leave a reading open or make it recoverable, and not both.

The script `verify_ch03.py` reproduces each figure of this chapter.

Part II

The Structure of an Edition

Chapter 4

Slots, Takes, and the Graph of Editions

The first chapter counted editions by multiplying matrices, and the method worked because every constraint tied a slot to its neighbor. A scene rarely behaves that way. A glass picked up in the second shot is still in the actor's hand in the fifth, whatever the inserts between them show, and a viewer who sees it change hands has found an error that no neighbor-to-neighbor rule can express. This chapter replaces the chain of constraints by a constraint structure in which a relation may join any set of slots. It gives a counting method that reduces to the matrix product when the structure is a chain, and it asks how the cost of counting and of ranking editions depends on the shape of the structure.

Two tests decide whether the generalization earns its place. The first is that, applied to the lawn scene of the first chapter with its chain of direction rules, the method must return exactly the totals already computed. The second is that it must handle a relation between two nonadjacent shots, namely the hand that holds a glass before and after an insert, and the example must show what a long-range relation costs and why. Both tests are run in the text and reproduced by the accompanying script. The chapter does not measure the width of any real scene, which is an empirical matter for the production that has one.

4.1 Constraint Structures

The setting is that of the first chapter. A scene plan has m slots numbered $1, \dots, m$. At slot j a production has a finite set K_j of perceptual classes of takes that pass the unary rules, and each class $k \in K_j$ has a weight $w_j(k)$, the number of takes in the class. A *class tuple* is a choice $\kappa = (\kappa_1, \dots, \kappa_m)$ of one class at each slot. The first chapter counted the class tuples that satisfy a chain of neighbor relations. The definition below drops the restriction to neighbors.

Definition 4.1 (Constraint structure)**.** A *constraint structure* on m slots consists of the sets $K_1, \dots, K_m$ together with a finite family $\mathcal{C}$ of *constraints*. A constraint is a pair $c = (\sigma_c, R_c)$ of a nonempty *scope* $\sigma_c \subseteq \{1, \dots, m\}$ and a relation $R_c \subseteq \prod_{j \in \sigma_c} K_j$. A class tuple κ is *admissible* if $\kappa|_{\sigma_c} \in R_c$ for every $c \in \mathcal{C}$, where $\kappa|_\sigma$ is the restriction of κ to the slots of σ. The *scope graph* of the structure has the slots as vertices and an edge $\{i, k\}$ whenever some scope contains

both i and k.

A constraint with a one-element scope is a unary rule and can be absorbed into the sets K_j, so the sets K_j already reflect it. Two examples are used throughout.

Example 4.2 (The chain)**.** The direction rule of the first chapter has one constraint $(\{j, j+1\}, R_j)$ for each $j < m$, where R_j contains the pairs of classes whose screen directions agree or one of which is neutral. The scope graph is a path on the m slots, and the number of admissible class tuples is the number N^{perc} of the first chapter. With the weights $w_j(k)$ the weighted count is the number N^{take} of take tuples.

Example 4.3 (The glass)**.** In the lawn scene the actor carries a glass of water and holds it in one hand whenever the actor is in frame. Slots 2 and 5 show the actor with the glass. Slots 3 and 4 are inserts that show the lawn and not the actor. A viewer will notice if the glass has changed hands across the inserts, so the production adds a *hand rule*. Each class at slots 2 and 5 carries an attribute $\eta \in \{l, r\}$, the hand that holds the glass, and the constraint $(\{2, 5\}, R_H)$ asks that $\eta(\kappa_2) = \eta(\kappa_5)$. This is a relation between two shots that are not neighbors. The scope graph is the path on five slots together with the extra edge $\{2, 5\}$, which closes a cycle through slots $2, 3, 4, 5$.

To have a definite example, the classes at slots 2 and 5 in the pools of the first chapter are given hand attributes alternately, left first, in the order in which the classes are listed at the slot. The labeling is arbitrary and is made only to fix the numbers. Table 4.1 gives the resulting counts.

	Slot 2: classes with hand l / r		Slot 5: classes with hand l / r	
Production	all classes	not stale	all classes	not stale
A	3 / 2	2 / 2	3 / 2	2 / 2
B	3 / 3	2 / 2	4 / 3	2 / 2
C	1 / 1	1 / 0	1 / 1	1 / 0

Table 4.1: Hand attributes of the classes at the two slots that show the glass. A class is stale if its takes show some strip in the wrong appearance, as in the first chapter.

4.2 Counting by Frontier Tables

The first chapter's matrix product keeps, at each slot, one number for every class at that slot. That is enough for a chain because the only information that the past passes to the future is the class at the latest slot. When a relation reaches over several slots the past passes more, and the method has to keep one number for every combination of what it passes. The *frontier* names what has to be kept.

Definition 4.4 (Frontier)**.** For $\ell = 0, 1, \dots, m$ the *frontier* Fr_ℓ is the set of slots $i \le \ell$ for which some slot $k > \ell$ shares a scope with i. The *frontier width* of the structure is $f = \max_\ell |\mathrm{Fr}_\ell|$.

The frontier Fr_ℓ consists of the slots already chosen that still have an unfinished relation with a slot to come. For the chain of Example 4.2, $\mathrm{Fr}_\ell = \{\ell\}$ for $0 < \ell < m$, and the width is 1. For the glass of Example 4.3 the frontiers are

$$\mathrm{Fr}_0 = \varnothing, \quad \mathrm{Fr}_1 = \{1\}, \quad \mathrm{Fr}_2 = \{2\}, \quad \mathrm{Fr}_3 = \{2, 3\}, \quad \mathrm{Fr}_4 = \{2, 4\}, \quad \mathrm{Fr}_5 = \varnothing,$$

and the width is 2. The slot 2 remains in the frontier across the inserts because its hand has not yet been compared with the hand at slot 5.

Lemma 4.5 (Frontier sufficiency)**.** *Let $x = (x_1, \dots, x_\ell)$ satisfy every constraint whose scope lies in $\{1, \dots, \ell\}$. Whether a completion $y = (y_{\ell+1}, \dots, y_m)$ makes (x, y) admissible depends on x only through $x|_{\mathrm{Fr}_\ell}$.*

Proof. Constraints whose scope lies in $\{1, \dots, \ell\}$ hold by hypothesis, and constraints whose scope lies in $\{\ell + 1, \dots, m\}$ do not involve x. A constraint whose scope meets both sides involves slots $i \le \ell$ that share the scope with a slot beyond ℓ, and every such i lies in Fr_ℓ by definition. Its satisfaction therefore depends on x only through $x|_{\mathrm{Fr}_\ell}$. □

Define for each ℓ a table T_ℓ on assignments z of classes to the slots of Fr_ℓ by

$$T_\ell(z) = \sum_x \prod_{i=1}^{\ell} w_i(x_i),$$

where the sum runs over the assignments x to slots $1, \dots, \ell$ that satisfy every constraint with scope in $\{1, \dots, \ell\}$ and satisfy $x|_{\mathrm{Fr}_\ell} = z$. The table starts as $T_0(\varnothing) = 1$.

Theorem 4.6 (Counting by frontier tables)**.** *The tables satisfy the recurrence*

$$T_\ell(z') = \sum_{(z,k)} T_{\ell-1}(z)\, w_\ell(k),$$

where the sum runs over the assignments z to $\mathrm{Fr}_{\ell-1}$ and the classes $k \in K_\ell$ such that every constraint whose largest slot is ℓ holds on (z, k) and $(z, k)|_{\mathrm{Fr}_\ell} = z'$. The weighted number of admissible class tuples is $T_m(\varnothing)$, and setting all weights to 1 gives the number of admissible class tuples. The number of table updates is

$$\sum_{\ell=1}^{m} |K_\ell| \prod_{i \in \mathrm{Fr}_{\ell-1}} |K_i| \;\le\; m\, q^{f+1}, \qquad q = \max_j |K_j|.$$

Proof. A slot $i < \ell$ that shares a scope with ℓ shares it with a slot beyond $\ell - 1$, so it lies in $\mathrm{Fr}_{\ell-1}$. Hence every constraint with largest slot ℓ has its other slots in $\mathrm{Fr}_{\ell-1}$, and it can be checked on (z, k). Also $\mathrm{Fr}_\ell \subseteq \mathrm{Fr}_{\ell-1} \cup \{\ell\}$, since a slot $i < \ell$ with a partner beyond ℓ has a partner beyond $\ell - 1$. These two facts make the recurrence well defined. Its correctness is induction on ℓ: an assignment to slots $1, \dots, \ell$ is admissible so far if and only if its

restriction to $1, \dots, \ell - 1$ is admissible so far and the constraints with largest slot ℓ hold, and by Lemma 4.5 the latter depends on the earlier slots only through the restriction to $\mathrm{Fr}_{\ell-1}$. At $\ell = m$ every constraint has been checked and the frontier is empty. Each update is one choice of (z, k), which gives the count. □

When the structure is a chain, the theorem returns the matrix product of the first chapter. The frontier $\mathrm{Fr}_\ell = \{\ell\}$ makes T_ℓ a vector indexed by the classes at slot ℓ, and the recurrence reads $T_\ell(k') = \sum_k T_{\ell-1}(k)\, A_{\ell-1}(k, k')\, w_\ell(k')$, which is the row-vector form of the product. The numbers confirm it. On the three productions of the first chapter the frontier method returns 494 and 238 perceptual editions for production A before and after the agreement rule, 320 and 96 for B, and 2 and 1 for C. It returns 120,042 and 57,834 take tuples for A, 10,240 and 3,072 for B, and 204,800,000 and 102,400,000 for C. All of these agree with exhaustive enumeration of the unweighted counts and with the totals of the first chapter. The first test is passed.

4.3 A Constraint Between Nonadjacent Shots

Add the hand rule of Example 4.3 to the chain. The tables must now carry, across slots 3 and 4, both the class at slot 2, which fixes the hand, and the class at the slot just chosen, which fixes the direction. The frontier at $\ell = 3$ is $\{2, 3\}$ and at $\ell = 4$ is $\{2, 4\}$, so the tables at those two stages are indexed by pairs of classes. For production A, after the agreement rule, there are 4 classes at slot 2, 3 at slot 3, and 3 at slot 4, so the two tables have $4 \cdot 3 = 12$ entries each, against at most 5 entries for the chain.

The counts with the hand rule are in Table 4.2. They agree with exhaustive enumeration. The frontier method needs 170 table updates for production A before the agreement rule and 102 after, against 86 and 58 for the chain alone.

	Perceptual editions		Take tuples	
Production	before agreement rule	after	before	after
A	241	119	58,563	28,917
B	160	48	5,120	1,536
C	1	1	102,400,000	102,400,000

Table 4.2: Admissible editions under the direction rule and the hand rule, before and after the agreement rule of the first chapter, with the hand labeling of Table 4.1.

Two features of the table are worth recording. The first is that a new rule can lower capacity before any viewer reports anything. With the labeling used here, production C, which supported two perceptual editions at the premiere under the direction rule alone, supports one once the hand rule is added, because its only stale class at slot 2 carries the hand r, and no class at slot 5 that is compatible in direction with its neighbors carries it. The

first chapter's result that a valid correction can break a promise was stated for corrections that follow a report, and the table shows that a rule adopted in advance has the same effect. The second feature is that the hand rule roughly halves the count for A and B. With two hands and alternating labels, a typical pair of classes at slots 2 and 5 agrees about half the time, and the exact counts are 241 of 494 for A before the agreement rule and 160 of 320 for B.

4.4 Width

The cost bound of Theorem 4.6 grows as q^{f+1}, so the width f decides whether counting is feasible for a scene with many slots. Three notions of width are in play, and they answer different questions.

4.4.1 Frontier width and the order of slots

The frontier of Definition 4.4 was defined for the screen order. Any linear order π of the slots defines frontiers in the same way, namely the slots that precede a cut in π and share a scope with a slot that follows it, and the corresponding width $f(\pi)$. The least value of $f(\pi)$ over all orders equals the pathwidth of the scope graph [8]. For the chain it is 1. For the glass the scope graph contains a cycle, and a cycle has pathwidth 2, so the screen order is optimal and no reordering of the slots reduces the frontier width below 2.

Remark 4.7 (Beyond a line). If the scope graph is tree-like and not path-like, as when two threads of action are intercut and each has its own internal continuity, counting by variable elimination along a tree decomposition of width t costs $O(mq^{t+1})$ [9, 10]. The treewidth is at most the pathwidth. Computing the treewidth is intractable in general [11] but takes time linear in the number of vertices when the width is fixed [12]. These are the guarantees to invoke when a scene's relations are not laid out along its screen order.

4.4.2 The cost of unrestricted structures

Width matters because without a bound on it the count is hard.

Proposition 4.8 (Hardness)**.** *Counting the admissible class tuples of a constraint structure is #P-hard, even when every set K_j has two classes and every constraint has a scope of two slots.*

Proof. Given a graph with vertices $1, \dots, m$, let $K_j = \{0, 1\}$ and for each edge $\{i, k\}$ add the constraint that $(\kappa_i, \kappa_k) \neq (1, 1)$. The admissible class tuples are exactly the indicator vectors of independent sets of the graph. Counting independent sets is #P-complete [13]. □

The proposition says nothing against the structures that arise in scenes, where relations come from objects and people that persist across a few cuts. It says that no method can be fast on all structures, and that the width is the parameter on which the method of Theorem 4.6 depends. A check on the claim for a small case is in the script: the frontier method counts independent sets correctly on random graphs of seven vertices.

4.4.3 Carry width

The frontier tables are indexed by the classes of the slots in the frontier, and for the glass that indexing is wasteful. The relation at slot 5 cares about the hand and not about which of the several classes at slot 2 carries it. What the future needs from the past is smaller than what the frontier stores. The earlier volume has the right notion, the quotient of histories by future equivalence, applied here to prefixes of editions [1, 7].

Definition 4.9 (Carry width)**.** A prefix $x = (x_1, \dots, x_\ell)$ is *viable* if it satisfies every constraint whose scope lies in $\{1, \dots, \ell\}$ and has at least one admissible completion. Two viable prefixes of length ℓ are *future-equivalent* if they have the same set of admissible completions. The *carry width at* ℓ, written c_ℓ, is the number of future-equivalence classes of viable prefixes of length ℓ, and the *carry width* of the structure is $c = \max_\ell c_\ell$.

Proposition 4.10 (Carry is at most the frontier table)**.** *For every* ℓ, $c_\ell \le \prod_{i \in \mathrm{Fr}_\ell} |K_i|$. *The carry width is the size of the smallest state space through which the counting and unranking of this chapter can be carried out, in the sense that any fold from viable prefixes that determines their admissible completions factors through the future-equivalence classes.*

Proof. By Lemma 4.5, viable prefixes with the same restriction to Fr_ℓ have the same admissible completions, so the classes of future equivalence are unions of the fibers of the restriction map, and there are at most as many classes as there are restrictions. The second statement is the proposition on fold sufficiency of the earlier volume, applied to the prefixes of admissible editions, in which a fold that determines futures has a kernel contained in future equivalence. □

The values for the lawn scene, computed by grouping all viable prefixes by their sets of completions, are in Table 4.3. Chain and glass are shown for productions A and B after the agreement rule, and production C has carry width 1 at every stage.

Production	Structure	$\ell = 0$	1	2	3	4	5
A	chain, carry width c_ℓ	1	3	3	2	2	1
A	chain, frontier table size	1	5	4	3	3	1
A	glass, carry width c_ℓ	1	3	4	4	4	1
A	glass, frontier table size	1	5	4	12	12	1
B	chain, carry width c_ℓ	1	2	2	2	2	1
B	chain, frontier table size	1	5	4	5	4	1
B	glass, carry width c_ℓ	1	2	4	4	4	1
B	glass, frontier table size	1	5	4	20	16	1

Table 4.3: Carry width and the size of the frontier table at each prefix length, after the agreement rule. The frontier table size is $\prod_{i \in \mathrm{Fr}_\ell} |K_i|$.

For the chain, the carry width is at most 3 in every production and at every stage, the number of screen-direction states, even though the frontier table at slot 1 has 5 entries. For the glass, the carry width at the stages across the inserts is 4 for both productions, while the frontier tables have 12 and 20 entries. The extra state beyond the chain is the hand. The carry width does not grow with the number of classes at slot 2, because every class with the same hand and direction behaves the same way for the future.

The saving can be stated in general for relations that act through an attribute.

Proposition 4.11 (Attribute compression)**.** *Consider a structure consisting of the direction chain and h constraints, each requiring that an attribute of slot a_t equal an attribute of slot b_t, where $a_t < b_t$ and the attributes take at most v values. At a cut ℓ crossed by h_ℓ of these constraints, meaning $a_t \le \ell < b_t$, the carry width satisfies $c_\ell \le 3v^{h_\ell}$.*

Proof. For a viable prefix of length ℓ, the admissible completions depend on the prefix only through the direction of its last take, which can take three values L, R, and N, and the values of the h_ℓ attributes of the slots a_t that the crossing constraints refer to. Prefixes that agree on these quantities have the same completions, and there are at most $3v^{h_\ell}$ combinations. □

The glass has $h_\ell = 1$ across the inserts and $v = 2$, so $c_\ell \le 6$. The exact value is 4, which is below the bound because not every combination of direction and hand occurs among the viable prefixes of these pools.

The advantage of the carry width over the frontier width shows when several relations cross the same cut. A synthetic scene of ten slots with four classes at each slot, a random direction at each class, and h nested relations that tie the attribute t of slot t to that of slot $11 - t$ for $t = 1, \dots, h$ gives the figures of Table 4.4. Each relation adds one slot to the frontier, which multiplies the frontier table by the number of classes, while each adds one attribute to the carry, which multiplies it by two.

Relations h	Width f	Frontier table	Attribute states	Updates
0	1	4	3	148
1	2	16	6	532
2	3	64	12	1,684
3	4	256	24	3,348

Table 4.4: A synthetic scene with $m = 10$ slots and $q = 4$ classes per slot, with nested long-range relations. The columns give the frontier width, the largest frontier table, the largest set of attribute states, and the table updates of the frontier method. Counts agree with exhaustive enumeration over all 4^{10} class tuples and with the counts of the attribute-state method.

The frontier table grows as q^f and the attribute states as $3 \cdot 2^h$, so the second is smaller as soon as q^{h+1} exceeds $3 \cdot 2^h$, which for four classes holds in every row of the table. The comparison shows what a long-range relation costs. It costs a factor q in the method of

Theorem 4.6, and when the relation acts through a small attribute it costs only the size of that attribute's range in a method built on the carry.

4.5 Counting and Unranking at Cost Set by Width

The first chapter described a pipeline in which a viewer receives a distinct integer, a keyed permutation sends it to another integer below the number of editions, and unranking turns that integer into a class tuple. Unranking was stated for the chain. The frontier tables make it work for any constraint structure.

Fix a total order on the classes at each slot. The *rank* of an admissible class tuple is its position, counted from 0, in the lexicographic order of the admissible class tuples.

Theorem 4.12 (Ranking and unranking)**.** *Let N be the number of admissible class tuples and f the frontier width. After building the completion tables $C_\ell(z)$, the number of admissible completions of a prefix of length ℓ whose restriction to Fr_ℓ is z, in at most mq^{f+1} table updates and with memory $\sum_\ell \prod_{i\in \mathrm{Fr}_\ell} |K_i| \le mq^f$ entries, the class tuple of rank $r \in [0, N)$ and the rank of a given class tuple can each be computed with $O(mqf)$ table lookups and constraint checks, whatever the value of f. The integers involved have at most $m \log_2 q$ bits.*

Proof. The tables are built backward from $C_m(\varnothing) = 1$ by the same recurrence as in Theorem 4.6, run from the last slot, and the bound on updates and memory is the same. To unrank r, choose the class at slot ℓ by walking through K_ℓ in its order. For each class that satisfies the constraints with largest slot ℓ given the choices so far, compare r with the table entry $C_\ell(z')$ for the resulting frontier assignment z'. If $r < C_\ell(z')$, choose that class, and otherwise subtract $C_\ell(z')$ from r and continue. Each of the m steps looks at most q classes and assembles a frontier key of at most f slots, which gives the bound. Ranking is the same computation read in the other direction, adding the table entries of the classes that precede the chosen one at each slot. The integer bound follows from $N \le q^m$. □

The theorem states how width enters the cost, and what it counts. The frontier bound controls the number of candidate table transitions. Total running time also depends on the cost of evaluating the constraints at each transition and on arithmetic with integers that can run to thousands of digits, neither of which the bound includes. It sets the preprocessing and memory, which grow as q^{f+1} and q^f, and it does not enter the per-viewer cost beyond the length of a frontier key. A production with a bounded width can therefore build its tables once per rule set and serve any number of viewers at a cost that depends only on the length of the scene. When the tables are built over the future-equivalence classes in place of the frontier assignments, and the transition function between classes has been tabulated, the memory is $\sum_\ell c_\ell$ entries and each slot of a viewer's unranking needs a constant number of lookups per class.

For the lawn scene the unranking is checked exhaustively. After the agreement rule, there are 238 class tuples for production A and 96 for B under the chain, and 119 and 48 under the

chain with the hand rule. In each case unranking every rank from 0 to $N - 1$ produces every admissible class tuple exactly once, in lexicographic order.

4.5.1 What an amendment costs

A rule that is added after a report changes the constraint structure, so the completion tables have to be rebuilt, and the cost of the rebuild is set by the width of the amended structure. A new relation between two slots adds an edge to the scope graph. If both endpoints already lie on the frontier at every cut that the edge crosses, the width does not change. If they do not, the width can grow by one, which multiplies the preprocessing by up to q. Amendments that tie slots that are far apart, such as the glass, are the expensive ones. The first chapter's agreement rule is unary and costs nothing in width.

Ranks are not stable across amendments. The lexicographic order of the admissible tuples changes when a rule removes some of them, and a viewer's rank under the old rules need not be the rank of the same edition under the new ones. The repair of the first chapter was therefore stated on class tuples, and it is the class tuple and not the rank that the ledger must record.

4.6 What the Chapter Establishes and Leaves Open

The chapter gives a counting method for editions under constraints between any slots, shows that it reproduces the totals of the first chapter when the constraints form a chain, and applies it to a relation between nonadjacent shots, the hand holding a glass across two inserts. It defines two widths, the frontier width of the scope graph and the carry width of future equivalence, and it shows how they bear on the cost of counting and unranking. It also shows that an unrestricted structure is intractable, so that the width is a property to be assumed and checked and not a convenience.

Three questions lead on from it. The first concerns the origin of long-range relations. In a real scene they come from objects and people that persist, and the earlier volume treats such agreement on overlaps by sheaf conditions. The next chapter takes a relation like the glass back to the overlaps from which it is derived, and asks how a correction that touches one overlap finds every other overlap it implicates. The second concerns the use of capacity. The methods of this chapter compute the number of editions that remain after a rule is added, and the cost of doing so after each amendment is a function of width. Whether a production can survive a family of amendments is the question of robustness, and whether its editions can be reached from the current assignments is the question of repairability, which counts alone do not answer. The third concerns the width of real scenes. The chapter's results are conditional on a bound on width, and what widths actual scenes have, and how they grow with the number of persistent objects, is a measurement that a production can make on its own footage.

The accompanying script `verify_ch04.py` reproduces every count of this chapter, including the chain totals of the first chapter, the counts with the hand rule, the carry widths

by exhaustive grouping of prefixes, the unranking of every rank, the independent-set check, and the synthetic scene.

Chapter 5

Continuity Beyond Neighbors

Chapter 4 counted editions under constraints between slots, and its glass example already showed a constraint between two shots that are not adjacent: the hand that holds a glass before an insert and the hand that holds it after. Chapters 9 and 10 then took amendments as given, each one a rule applied at a stated set of slots. This chapter supplies what both left open. It says where continuity constraints come from, how they are written as the constraint structures of Chapter 4 with the cheapest width, and, when a viewer reports a continuity defect, which slots the resulting amendment has to touch.

The organizing idea is that continuity is a property of an entity across the showings in which it appears, and that an event in the story changes an entity at one moment and leaves it alone before and after. The slots that an amendment must touch are the showings of the entity within a stretch that no event interrupts. The chapter derives from this the three slot sets used as running examples in Chapter 9, $\{2, 3, 5\}$ for strip a, $\{3, 5\}$ for strip b, and $\{1, 4, 5\}$ for strip c, and it shows why the third is the union of two stretches.

5.1 Entities, Events, and Showings

A story world has *entities*, the things whose appearance a viewer can track from shot to shot: the strips of the lawn, a glass, a coat. It has *events*, the occurrences of the script that change entities, each with an *effect set*, the entities it changes, and a time. A *showing* is a pair of a slot and an entity visible in it. The slots of the lawn scene are those of the first chapter, with times $0, 1, 2, 2, 3$ and visible strips $\{a, b, c\}$, $\{a\}$, $\{a, b\}$, $\{c\}$, $\{a, b, c\}$. The script mows strip a between times 0 and 1, strip b between 1 and 2, and strip c between 2 and 3.

Definition 5.1 (Persistence interval and footprint)**.** For an entity e, let the events whose effect sets contain e have times $t_1 < \cdots < t_k$. The *persistence intervals* of e are the maximal stretches of time that contain none of these event times in their interior, namely $(-\infty, t_1)$, $[t_1, t_2)$, and so on. The *footprint* $F(e)$ is the set of slots that show e, and the *interval footprint* of e at a slot s is the set of slots of $F(e)$ whose times lie in the same persistence interval as the time of s.

For the lawn each strip has one event, so two intervals, before and after the mowing. The

showings and intervals are in Table 5.1. Strip a is shown in slots $1, 2, 3, 5$, once before it is mown and three times after. Strip c is shown at slots 1 and 4 before and only at slot 5 after. The *late footprint* of Chapter 1 is the interval footprint after the event.

Strip	Footprint F	Before its event	After its event
a	$\{1, 2, 3, 5\}$	$\{1\}$	$\{2, 3, 5\}$
b	$\{1, 3, 5\}$	$\{1\}$	$\{3, 5\}$
c	$\{1, 4, 5\}$	$\{1, 4\}$	$\{5\}$

Table 5.1: Showings of each strip and the two persistence intervals that its mowing event creates.

5.2 Absolute and Relative Continuity

Two kinds of continuity constraint arise, and they are written differently.

Absolute continuity compares a showing with the script. The script says what a strip looks like at each time, mown or unmown, and a take that shows the strip otherwise is stale in the sense of Chapter 1. The constraint has a scope of one slot, so it restricts the domain of that slot and creates no edge in the scope graph. The agreement rule G is the conjunction of these constraints.

Relative continuity compares showings with each other, for an attribute that the script does not fix. The hand that holds a glass is the standing case: the script does not say which hand, and the requirement is only that the glass is in the same hand in every showing within a persistence interval. Such a constraint has a scope of several slots and an equality relation. Across an event that is declared to pass the glass from one hand to the other, the relation between the showings on the two sides is the opposite of equality. Both are relations in the sense of Chapter 4, and the equalities are the overlap conditions of the sheaf of takes in the earlier volume [1]: two takes that show the same entity must agree on the overlap where both show it.

5.2.1 Path and Clique Representations

Within a persistence interval an entity shown at k slots can be constrained in two natural ways. The *clique* representation writes an equality between every pair of showings, $k(k-1)/2$ binary relations. The *path* representation writes one between each pair of consecutive showings, $k-1$ relations.

Proposition 5.2 (The path suffices)**.** *Equality is transitive, so the path and clique representations of the same interval have the same solutions, and the same holds for any relation family that composes along the path, such as an opposite relation across a pass-over event. The frontier width of the path form is at most the frontier width of the clique form, and the difference can be unbounded.*

Proof. If consecutive showings agree then all agree, which gives the first statement, and the clique constraints are implied by the path. The frontier at a cut of the scope graph is the set of slots at or before the cut that share a relation with a later slot. Every path edge is in the clique form, so every slot live in the path form is live in the clique form, giving the inequality. For the unboundedness, take an entity shown at every slot of an n-slot chain. The path edges coincide with chain edges and the frontier width stays at 1, while the clique form makes all earlier slots live at every cut, a width of $n - 1$. □

The lawn glass shows the same effect on a small scale. Let the glass appear at slots 1, 2 and 5 and let its hand alternate between left and right in the listing order of the classes of each slot, as in Chapter 4. The path form has relations on slots $\{1, 2\}$ and $\{2, 5\}$ and the clique form adds $\{1, 5\}$. The solution counts agree, as the proposition says: 125 class tuples for A, 84 for B and 1 for C under the direction rule, and 57, 24 and 1 once the agreement rule G is added. The frontier width is 1 for the chain, 2 with the path edges, and 3 with the clique. Table 5.2 gives the synthetic case, in which the table of partial assignments at a cut has q^w entries for q classes per slot and width w, so that for $q = 4$ and $n = 8$ the path form has 4 entries and the clique form 16,384.

n	Path width	Clique width	Path table ($q = 4$)	Clique table ($q = 4$)
3	1	2	4	16
4	1	3	4	64
5	1	4	4	256
6	1	5	4	1,024
7	1	6	4	4,096
8	1	7	4	16,384

Table 5.2: One entity shown at every slot of an n-slot chain, in path and clique form.

An event with an effect set that passes the glass from one hand to the other changes the relation and not the representation. Suppose the script has the glass change hands between slots 2 and 5. The relation on $\{2, 5\}$ becomes the opposite of equality and the relation on $\{1, 2\}$ stays equality. For A the counts are 122 and, with G, 57. For B they are 76 and 24. Production C falls to 0: the only admissible classes at slots 2 and 5 after G both carry the left hand, so no edition of C satisfies the opposite relation. The production would need a footage change before the script could be filmed this way.

5.3 From a Defect to the Slots It Touches

A viewer reports a defect: in slot j the entity e appears otherwise than it should. The report is adjudicated as in Chapter 7, and if it is upheld as evidence for an amendment, the amendment has to say which slots it governs. The rule of this chapter is that it governs the persistence interval of the defect.

Definition 5.3 (Touched slots)**.** An *absolute defect* is a pair (e, j) such that a class at slot j shows e in an appearance that differs from the script's. The slots *touched* by an amendment answering it are the interval footprint of e at j. A *relative defect* is a pair $(e, \{j, j'\})$ of two showings of e in one persistence interval that disagree on a free attribute, and the slots touched are the interval footprint of e at j, which contains j'.

The rule is sound in that it never crosses an event boundary. A strip that is mown at time 1 is shown unmown before it, correctly, and an amendment about the reappearance of grass after the cut must not remove the unmown classes of earlier shots. It is complete in the following sense.

Proposition 5.4 (Soundness and completeness)**.** *Let e be an entity and I a persistence interval of e. A class at a slot of $F(e)$ can be stale for e in the way a defect in I shows only if its slot lies in I, because the required appearance is constant on I and a class is judged against the requirement at its own time. The requirement outside I may happen to equal the value on I, since an attribute can return to an earlier value, so the interval is the declared scope of the repair and not a claim that the requirement differs elsewhere. Consequently the amendment $\alpha(\{e\}, J)$ with J the interval footprint of a defect removes every class that is stale for e in that interval and no class that is correct for e at its slot. For a relative defect, the amendment that imposes the interval's equality relation on J rules out the disagreeing pair and every other disagreement within I. That is a continuity policy for the whole interval. The smallest amendment that blocks the reported witness is a condition on the pair (j, j') alone, and it leaves other pairs in I free to disagree.*

Proof. The script fixes the required appearance of e at each time, and it changes only at event times of e. A class at a slot in I is stale for e exactly when its displayed appearance differs from the value that holds on all of I. A class at a slot outside I is compared with the requirement at its own time, and removing it for this defect would remove a class that is correct there. For the relative case, a pair disagreeing on a free attribute is ruled out by an equality relation on any scope containing both showings, and the interval footprint is the scope that covers every pair in the interval. A condition on the reported pair alone also rules out that pair, with a smaller scope and a weaker guarantee. □

5.3.1 The Three Slot Sets of Chapter 9

The stale classes of Table 9.2 of Chapter 9 determine the slot sets of its amendments. Every stale class of strip a, in every production, lies after a's mowing, at slots 2 and 3, so its defect touches the interval footprint $\{2, 3, 5\}$. The stale classes of strip b are at slots 3 and 5 and touch $\{3, 5\}$. Strip c is the interesting case. In production A its stale class is at slot 4, which shows c already mown at time 2, before the script mows it. That is a premature defect, in the interval before the event, with interval footprint $\{1, 4\}$. In productions B and C the stale classes are at slot 5, after the event, with interval footprint $\{5\}$. The amendment that answers all the reported defects on c is the union $\{1, 4\} \cup \{5\} = \{1, 4, 5\}$, which is the whole footprint and the slot set of α_c. Applying all three derived amendments is the type-level rule G on every pool, which was checked.

The distinction matters in the counts. Table 5.3 gives the capacity after the amendment on each interval of c and after the entity-level amendment. For production A the premature defect lies in the interval $\{1, 4\}$ and the late interval alone removes nothing, leaving 494 class tuples; for B the opposite holds and the early interval alone removes nothing, leaving 320. An amendment that confused the two intervals would repair the wrong half.

Production	Premiere	c before $\{1, 4\}$	c after $\{5\}$	c entity $\{1, 4, 5\}$
A	494	422	494	422
B	320	320	176	176
C	2	2	2	2

Table 5.3: Capacity after the amendment on each interval of strip c, and on its whole footprint.

5.4 Levels of Amendment

An amendment can govern the interval of the defect, every showing of the entity, or every entity of the same kind. These are nested.

Definition 5.5 (Levels)**.** For an absolute defect (e, j) the *interval level* is the interval footprint of e at j, the *entity level* is $F(e)$, and the *type level* is the set of all slots, with the agreement rule applied to every entity of the type. Then interval $\subseteq$ entity $\subseteq$ type.

The loss grows with the level, and so does the protection. For strip a the interval level $\{2, 3, 5\}$ and the entity level $\{1, 2, 3, 5\}$ differ only by slot 1, where no class shows a in a stale appearance, so their capacities are equal in every production: 351, 200 and 1 for A, B and C. The same holds for strip b with 404, 288 and 2. Strip c separates the levels, as above. The type level is G and leaves 238, 96 and 1.

The choice of level is a decision about evidence. An interval amendment responds to a defect that has been seen, and leaves unconstrained a stale class that could be reported next. An entity amendment closes every interval of the entity at once. A type amendment treats the report as evidence of a rule for the whole production. Chapter 8 describes the decision, and its result bears directly on the capacity of Chapter 9: the lower the level, the smaller the loss, and the more correction events the production must expect.

5.5 Coupled Entities

An event whose effect set contains several entities couples them. If the script has strip b mown at the same moment that the glass changes hands, then moving that event in time moves the change of hand along with it, and a defect about the time of the event concerns both. The footprint of an event is therefore the union of the footprints of the entities in its effect set. Here that is $F(b) \cup F(\text{glass}) = \{1, 3, 5\} \cup \{1, 2, 5\} = \{1, 2, 3, 5\}$, one slot more than

$F(b)$ alone. The coupled amendment on strip b at slots $\{1, 2, 3, 5\}$ leaves the same capacity as the entity amendment, 404, 288 and 2, because no class at slot 2 misdepicts b, so the extra slot costs nothing in this pool. In a pool with such a class it would.

The closure rule is the general one. Starting from the entity of the defect, take the interval footprint; add every entity that shares an event with the entity in that interval; take their interval footprints in turn; repeat until nothing is added. The closure is finite because there are finitely many entities and intervals. It is sound in that it adds a slot only through a shared event, and so it does not reach entities that merely appear in the same shot.

5.6 What the Chapter Gives the Others

Chapter 4 gave the cost of counting editions in terms of width, and this chapter shows how the width of a real scene is fixed by how its continuity is written: a path through the showings of an entity keeps the frontier at one slot per live entity, and a clique makes it grow with the number of showings. Chapter 9 took its slot sets as given, and this chapter derives them from the pools and the script. Chapter 10 took its pools as given as well, and the level of the amendment now bears on its repair problem: a lower level removes fewer classes, so the pool is larger and fewer viewers hold an invalid edition, at the price of more amendments over time.

The scripts `lawnmodel.py` and `verify_ch05.py` reproduce each figure of this chapter, including the equality of path and clique counts, the widths of Table 5.2, the derivation of the slot sets, and the capacities of Table 5.3.

Chapter 6

Sound, Dialogue, and Time

Image and sound do not occupy a film in the same way. A shot can be replaced by another of the same class without moving anything else, and the counts of the earlier chapters have treated slots as if that were always so. Sound occupies time. A line of dialogue takes a certain number of seconds, a music cue lasts until it ends, and a stretch of loud material either leaves room for the listener to recover or it does not. A repair that fits the picture can break the soundtrack, because the picture's repair changes how long something takes.

This chapter extends the model of editions from discrete classes to continuous timing. A slot gets a duration, constrained by the takes and by the sound anchored to them, and an edition is a class tuple together with a timing. The chapter then takes the analysis of acoustic occupation in an existing film as a second worked case, with measured data, and asks how much a cut can be shifted before a recovery window or an occupation block changes. It ends with the asymmetry that a saturating response gives to those tolerances, which is the reason the tolerance of a cut to a timing shift is a different kind of quantity from the tolerance of a class to a substitution.

6.1 Timing as a Second Kind of Choice

Each slot of a scene plan has a duration τ_j in seconds. A take has a length and some handle at either end that lets an editor trim it, so a slot has a feasible interval $[\ell_j, u_j]$ of durations, the lower end set by the least that the content needs and the upper end by how much footage the take offers. A line of dialogue anchored to the slot raises the lower end, since the line has to fit.

Definition 6.1 (Timed edition)**.** A *timed edition* is a class tuple together with a duration vector $\tau = (\tau_1, \dots, \tau_m)$ in the feasible region $\mathcal{T} \subseteq \prod_j [\ell_j, u_j]$. The feasible region is the box of the slot intervals cut by any constraints that tie durations together, the commonest being that the total duration of the scene is fixed so that the scene stays aligned with the reel around it.

Two timings are not distinguishable to a viewer if they differ by less than a tolerance ε_t, and this is the timing analogue of the tolerance of Chapter 2. For a slot with an interval of

length $u_j - \ell_j$ the number of timings that a viewer can tell apart is about $\lfloor (u_j - \ell_j)/\varepsilon_t \rfloor + 1$ if two timings count as different when they differ by at least ε_t. Over the whole scene the timings that can be told apart form a packing in $\mathcal{T}$. The distance between two duration vectors is taken here as the largest difference in any slot, and two timings count as different when that distance is at least ε_t. This convention differs at the boundary from the packing of Chapter 2, which asks for separation strictly greater than the tolerance, under which a grid spaced exactly at the tolerance would not be a packing. The counts of this chapter are for the convention of at least, and are not directly comparable with a packing number of Chapter 2.

To see the scale, give the five slots of the lawn scene nominal durations of 4, 3, 5, 3 and 6 seconds and a handle of one second either way, and take a tolerance of half a second. Each slot has five distinguishable durations and the box has $5^5 = 3{,}125$ timings, which is 11.6 bits. If the total duration is held to its nominal 21 seconds, the number of timings drops to 381, or 8.6 bits, since a shot that runs long must be paid for by another running short. Now let the second and fourth slots hold lines of dialogue that cannot be trimmed below their nominal length. Their intervals shorten to one second, the box falls to 1,125 timings, and with the total fixed there are 126, or 7.0 bits.

Constraint on the timings	Distinguishable timings	Bits
Handles of ± 1 s, no other constraint	3,125	11.6
Total duration fixed	381	8.6
Dialogue at slots 2 and 4 cannot be trimmed, no other constraint	1,125	10.1
Dialogue at slots 2 and 4, total fixed	126	7.0

Table 6.1: Grid timings of the lawn scene at tolerance 0.5 s. These count points of a grid with spacing equal to the tolerance, so they are lower bounds on the packing numbers of the same regions.

The timing bits add to the class bits under two assumptions: that the feasible region of timings is the same for every class tuple, and that two timed editions count as different when their classes differ or their timings differ by at least the tolerance, which is a joint comparison rule and not a consequence of the two separate ones. In an actual edit the handles and the dialogue anchors belong to takes, so the feasible region can depend on the take chosen, and the product of the two counts is then an upper bound on what the independence assumption gives and not a count of the edits available. Production A has 7.9 bits from its 238 classes after the agreement rule and could have 7.0 more from timing, giving 29,988 timed editions. Production B has 6.6 and 12,096. Production C has none from its classes and the same 7.0 from timing, 126 timed editions. Timing can therefore supply distinguishability that the footage cannot. Whether it does depends entirely on the tolerance: if viewers cannot tell a cut that is half a second late from one that is on time, the extra bits are zero, and if they can, production C has a hundred and twenty-six perceptibly different editions from a single class tuple. The tolerance of half a second is assumed. It is not a measurement.

6.2 The Acoustic Occupation of an Existing Film

The measured case is the analysis of the soundtrack of a 2026 action film, *Drive Through Fire*, in the monograph on acoustic occupation [14]. The analysis computes the momentary loudness of the film's audio second by second, takes as a reference the loudness of the seconds that subtitles mark as dialogue, and measures how far the rest of the soundtrack stands above it. The film's per-second table has 5,222 rows. The dialogue reference is -34.6 LUFS, and the integrated loudness of the film is -21.6 LUFS.

The analysis defines two structures, both under stated parameters. A second is *occupied* if its loudness exceeds the dialogue reference by at least $\delta = 10$ dB. An *occupation block* is a run of occupied seconds, with gaps of at most five seconds merged, lasting at least a minute. A *recovery window* is a stretch of at least five seconds in which the loudness is at or below the dialogue reference, or the soundtrack is inaudible. The film has nine occupation blocks, totalling 16.7 minutes, the longest 4.7 minutes. It has 105 recovery windows, totalling 1,526 seconds, with a median length of 11 seconds.

The passage from 1:00:39 to 1:20:51, a span of 1,212 seconds bounded by the longest stretches without a qualifying window, contains six of the blocks and three recovery windows, of 9, 7 and 13 seconds, 29 seconds in all. The last two windows are separated by a single second above the reference.

These figures are acoustic opportunities for recovery under the stated rules. They are not measurements of a listener's recovery, which would require listeners, and the parameters δ, the merge gap, the minimum block length and the window length are the analysis's choices and have not been validated on an audience. The film is one film.

6.3 Tolerance of a Cut to a Shift

A repair that changes the length of a shot changes the length of whatever stretch of sound it falls in. The measured structures say how much each can lose before it changes.

A recovery window of length w seconds stops being a window when it falls below five, so a cut inside it can remove at most $w - 5$ seconds, and the window's slack is that amount. Over the film the 105 windows have a total slack of 1,001 seconds. The slack is far from evenly spread. Twenty-three of the windows are five or six seconds long, within a second of ceasing to qualify. In the passage the three windows have slack of 4, 2 and 8 seconds, 14 in all, and a cut that removed the single second between the last two windows would merge them into one window of 21 seconds and reduce the number of windows in the passage from three to two.

An occupation block of duration b seconds stops being a block when it falls below the minimum of sixty, so its slack is $b - 60$. The nine blocks last 72, 62, 111, 101, 92, 79, 130, 73 and 284 seconds, with slack of 12, 2, 51, 41, 32, 19, 70, 13 and 224. One block has a slack of two seconds. The gaps between consecutive blocks matter in the other direction. Two blocks merge into one if the gap between them falls to the merge gap of five seconds. The gaps in

the film are 641, 910, 1,354, 115, 50, 79, 131 and 11 seconds, and only the last is within reach of a short cut: removing six or more seconds from the eleven-second gap that ends at 1:15:09 would merge the two blocks on either side, which last 73 seconds and 284 seconds, into a single block of about six minutes. The count of blocks would fall from nine to eight.

Proposition 6.2 (Slack)**.** *Let a structure be defined by a threshold on length, so that it exists when a stretch has length at least L. A cut that shortens the stretch by s preserves the structure if and only if $s \le w - L$, where w is the present length, and lengthening never destroys it. For a structure defined by a gap at most g that merges neighbors, a cut that shortens the gap by s preserves the separation if and only if $s < w - g$, and lengthening the gap never merges.*

Proof. Both statements restate the thresholds. A stretch of length $w - s$ is at least L exactly when $s \le w - L$, and a gap of length $w - s$ exceeds g exactly when $s < w - g$. Lengthening increases the stretch and the gap respectively. □

The proposition gives the asymmetry that the analysis exposes. Shortening and lengthening are not mirror images. A window can be lengthened without limit and shortened by its slack, and a block likewise, and a gap can be lengthened without limit and shortened only until it merges the blocks beside it. A timing shift that moves content later is a lengthening of whatever lies before it and a shortening of whatever lies after, and so it is permitted by the sound only within the slack of both.

6.4 Saturating Responses and Asymmetric Tolerance

The slack above is about whether a structure survives. A second asymmetry concerns how much a viewer notices. The earlier volume's stipulated response model for a pause before a character's question has the anticipation that the pause builds as $q_1 = \tanh\big((p_1 + \tfrac{1}{2}p_4)/6\big)$, in which p_1 is the pause in seconds and p_4 is the duration of the reaction shot. The function is a stipulation chosen for transparency, and it is not fitted to viewers. At the base values $p_1 = 7$ and $p_4 = 4$, the anticipation is 0.9051.

For a tolerance of 0.05 in anticipation, the linearization predicts a symmetric tolerance of ± 1.66 seconds. The model itself says that the pause can be shortened by 1.35 seconds, to 5.65, and lengthened by 2.32 seconds, to 9.32. A reduction of four seconds is predicted by the linearization to cost 0.120 and costs 0.223. The reason is that tanh is concave: it flattens as the pause grows, so extra pause buys little, and it steepens as the pause shortens, so a cut costs more than the slope at the base point suggests.

Proposition 6.3 (Asymmetry of a saturating response)**.** *If a response q is increasing and concave in a timing p, then the tolerance region $\{p : |q(p) - q(p_0)| \le \varepsilon\}$ is not symmetric about p_0 unless q is linear on it: the shortening that is tolerated is shorter than the lengthening that is tolerated.*

Proof. Concavity gives $q(p_0 - h) \le q(p_0) - hq'(p_0)$ and $q(p_0 + h) \le q(p_0) + hq'(p_0)$ with strict inequalities unless q is linear. The first says that a shortening by h costs at least $hq'(p_0)$

and the second that a lengthening by h gains at most that. A lengthening by h therefore stays within ε of the base for at least as long as the shortening does, and for longer when q is strictly concave. □

The two asymmetries have different sources and different consequences. The slack of Proposition 6.2 is a fact of the soundtrack and is the same for every viewer: a window of seven seconds can lose two. The asymmetry of Proposition 6.3 is a property of a response model, and it depends on the viewer and on how the model is fitted. A cut that has to be shortened is limited by both, and the limit that binds is the smaller.

6.5 What the Chapter Gives the Others

The extension to continuous timing changes what a slot is. Chapter 4's constraint structures had classes at slots and relations between them, and a timed edition adds a coordinate at each slot with constraints that tie the coordinates together. The width arguments of that chapter carry over, since a constraint on durations between two slots is a relation between them, and the cost of counting depends on the same frontier. What changes is that the domain at each slot is an interval and the relations are inequalities, and the counting of distinguishable timings is a packing in a polytope.

Chapter 8's footage and relaxation options gain a third: a correction can be made by changing timing, within the slack the sound allows, and this costs no new footage. A repair that needs a shot to run longer by two seconds is formable when some structure in the scene has two seconds of slack to give, and not otherwise. Chapters 9 and 10 count capacity and repair radius in class tuples. A timed edition has a larger space in which to repair, and a viewer whose class is unavailable might be moved in time and not in class, at a disturbance that is measured in the timing tolerance.

The limits of the chapter are the limits of its inputs. The timing tolerance of half a second is an assumption. The measured structures are from one film, defined by stated parameters, and they are opportunities and not recoveries. The response model is stipulated. The propositions hold for any thresholds and any concave response, and the numbers show what they look like for these.

The scripts `verify_ch06.py`, with the per-second table and block list that it reads, reproduce each figure of this chapter.

Part III

Correction

Chapter 7

Reports and Adjudication

A viewer reports that the grass has reappeared after the cut. Before any correction, the production has to decide what the report is worth. The viewer may have described the film wrongly, the film may be right and the viewer’s inference mistaken, or the film may really contain the defect, in which case the defect may be a failure to follow a rule already in force or a sign that the rules are incomplete. Chapter 1 gave a three-step verdict for this decision. This chapter develops it into an account of reports: what the ledger must hold for the verdict to be reached, how the edition that was assigned differs from the edition that was played, which defects a report can describe, what many reports about one slot add, and why the production must keep the footage that it withdraws.

The chapter uses the lawn scene and the audiences of Chapters 9 and 10, and its counts come from the same pools. Production C yields a result that is worth stating at the outset. Its two perceptual editions differ at one slot only, the second slot, and the class that makes them differ is a stale class. The second edition is the edition with the continuity error. The production’s capacity of two before the correction consists of one correct edition and one defective one, and what a report from the viewer of the second edition sets in motion is the loss of the variation.

7.1 What the Ledger Holds

The assignment ledger records, for each viewer, the revision of the production at which the edition was assigned and the take tuple assigned. A revision is the pair of the rules in force and the pool of takes at a time. Two things are therefore recoverable from the ledger for any viewer: the take that the production assigned, and the rules under which it was admissible.

The edition that the viewer actually played is a separate matter. A viewer may play a cached copy of an earlier revision, an offline file, or a segment that streamed before a repair was pushed. The ledger holds the assignment and cannot show the playback. Playback has to report a *revision identifier*, which names the revision of the file that was played, and the identifier together with the take tuple of that revision fixes the played edition. When the two differ, the verdict is reached against the played edition.

7.2 The Verdict

A report is a *claim* (j, s, ap): at slot j, strip s appears in the appearance ap, mown or unmown. The script requires an appearance of s at each time, constant between its events, and the played take shows some appearance of s at slot j, which differs from the requirement exactly when the take's class is stale for s.

Definition 7.1 (Verdict)**.** Let shown be the appearance of s at slot j in the played take and req the appearance that the script requires. The report is *unsupported* if $\mathsf{ap} \neq \mathsf{shown}$. It is *rejected* if $\mathsf{ap} = \mathsf{shown} = \mathsf{req}$, so that the viewer saw correctly and nothing is wrong. It is *upheld* if $\mathsf{ap} = \mathsf{shown} \neq \mathsf{req}$. An upheld report is a *violation* if the played take's class was inadmissible under the rules in force at its revision, and *evidence for an amendment* if the class was admissible under them.

The verdict is a function of four things that the ledger and the script supply: the claim, the played take, the script's requirement, and the rules at the revision of play. No judgment is needed beyond comparison.

Proposition 7.2 (Trichotomy)**.** *Every claim receives exactly one of the outcomes unsupported, rejected, upheld as a violation, or upheld as evidence for an amendment, whenever the played take and the rules at its revision can be recovered, and the outcome is then computed from the ledger entry of the played revision and the script. When they cannot be recovered the claim is* unjudged*. That outcome is a fact about the ledger and not about the film, and it differs from a claim that fails verification, which is unsupported. A claim about a showing is unsupported for exactly one of the two appearances, and for the other it is rejected or upheld according as the played take is correct or stale at that showing.*

Proof. Exactly one of $\mathsf{ap} = \mathsf{shown}$ and $\mathsf{ap} \neq \mathsf{shown}$ holds, since there are two appearances, and in the first case shown equals req or does not. In the upheld case the rules at the revision either exclude the class or admit it. □

To see the proportions, take the premiere audiences of the lawn scene and file every possible claim about every showing from every viewer. Production A has ten showings, so twenty claims per viewer and 1,200 in all for its audience of sixty. Of these, 600 are unsupported, as the proposition says half of them must be, 563 are rejected, and 37 are upheld. The 37 upheld claims are the stale showings that the audience actually holds. They are evidence for an amendment if the audience was assigned under the direction rule alone, and they are violations if the rules in force already included the agreement rule, in which case the assembly delivered an inadmissible edition. Table 7.1 gives all three productions.

The count of unsupported claims is a feature of the setup and says nothing about viewers' accuracy: it is the number of showings times the number of viewers, because exactly one of the two claims about a showing agrees with the take. Real reports are not exhaustive claims, and the exhaustive count serves to show how few of the possible claims can be upheld.

Production	Claims	Unsupported	Rejected	Upheld
A (60 viewers)	1,200	600	563	37
B (30 viewers)	600	300	274	26
C (2 viewers)	40	20	19	1

Table 7.1: Every possible claim about every showing, adjudicated against the premiere assignment. The upheld claims are amendment evidence under the direction rule and violations under the agreement rule.

7.3 Assigned Versus Played

Suppose a correction is made, the invalid viewers are repaired as in Chapter 10, and a viewer plays an old file. The viewer reports a stale class that they saw, and the report is judged against the ledger's current assignment. The assigned take is the repaired one, which is correct at that showing, and the claim does not match it.

Proposition 7.3 (Judging the played edition)**.** *Let a viewer's played revision differ from the assigned revision, and let the two takes show different appearances of the reported strip at the claimed showing, so that they disagree on the reported feature and not merely in framing or grain. A truthful claim about the played take is unsupported when judged against the assigned take, and is upheld when judged against the played take, if the played take was stale there.*

Proof. A truthful claim matches the played take, and the assigned take shows a different appearance at that showing, which makes the first comparison fail. The second comparison is the definition of upheld. □

The lawn scene gives the size of the effect. After the agreement rule is adopted, the 31 invalid viewers of production A and the 20 of production B are repaired to correct editions. If each of them reports the stale class that they saw at the earlier revision, then judged against the assigned revision all 31 and all 20 reports are unsupported, and judged against the played revision all of them are upheld. A production that adjudicates against assignments alone will therefore dismiss the truthful reports that are most likely to follow a correction, from the viewers who still have the old file.

7.4 A Taxonomy of Defects

The kinds of defect that a report can describe correspond to the constraints of Chapters 4 and 5. An *absolute defect* is a showing that departs from the script. It comes in two kinds: a *reappearance*, in which an entity shows its state from before an event after the event, and a *premature* defect, in which it shows its state from after an event before the event. A *relative defect* is a disagreement between showings of an entity on a free attribute, such as a glass in different hands. A *direction defect* is a violation of the direction rule between consecutive

slots, which concerns the edit and not a depicted entity. A *description error* is a report that does not match the played take, which the verdict marks as unsupported.

The distinctions matter for what follows from an upheld report. An upheld absolute defect has an interval footprint by Proposition 5.4 of Chapter 5. A relative defect leads to an equality or opposite relation on the showings in an interval. A direction defect is always a violation of a rule that every production has in force, and the remedy is repair and not amendment. A description error leads to nothing, though a pattern of description errors about one slot is a sign that the slot is ambiguous to viewers, which is information of another kind.

7.5 Many Reports About One Slot

Two viewers who hold the same stale class report the same defect. The class is the unit of evidence, because the take variants within a class are perceptually equivalent and show the same defect. Audience size therefore controls what can be found out. In the audience of production A, the four stale classes of the pool are held by 6, 6, 10 and 15 viewers, thirty-seven viewer-holdings in all, and every stale class in the pool has been reached. In B, the seven stale classes are held by between 1 and 6 viewers each, and again every stale class is reached. In C the audience has two viewers, one of whom holds a stale class and one of whom holds none, so one of the two stale classes of the pool is held, and the defect on strip c at the last slot is held by nobody.

The same view applies at the level of the amendment. The reports of A reveal a defect on strip a after its event, on strip b after its event, and on strip c before its event, which is every pair of entity and persistence interval that the pool contains. The reports of B reveal all three of its pairs. The audience of C reveals one of its two: strip a after its event, and not strip c. An amendment derived from the reports of C would repair the defect on a and leave a defect on c that no viewer has yet been shown.

This is the point at which the standard of proof for a rule becomes concrete. A single verified instance is enough for the interval-level amendment by Proposition 5.4 of Chapter 5, since it removes the stale classes of that entity in that interval and nothing else. It is not evidence about the other intervals of the entity, or about other entities, and a production that wants to adopt the entity-level or type-level rule on one instance has made a decision that the instance does not support. Chapter 8 treats the choice.

7.6 What Must Be Retained

A verdict about a past edition can be reached only if the played take can be identified and recovered, and if the rules at its revision can be reconstructed. The first requires that the assets themselves be kept after they stop being assigned.

The take-level amendment ω of Chapter 9 withdraws one take from every class, and it was shown to cost no perceptual capacity. For adjudication it is not free. The premiere

audience of production A holds 68 of the 69 takes in the pool, so almost the whole pool has been played by someone. Take variant 0, which ω withdraws from each class, appears in 96 of the 300 assigned take slots of the audience of A, in 72 of 150 for B and in 1 of 10 for C. Fifty-six of A's sixty viewers hold at least one such take. If the withdrawn takes are deleted, a report from any of them about their premiere edition cannot be judged, and the production loses the ability to distinguish a truthful claim from a mistaken one.

Proposition 7.4 (Retention)**.** *A verdict about the edition that a viewer played at a past revision can be reached if and only if the take tuple of that revision and the rules at that revision can both be recovered. A withdrawal that deletes an assigned take makes every verdict about the viewers who held it unreachable. A withdrawal that moves the take to an archive, out of the admissible pool but still recoverable, does not. Claims that become unjudged by such a withdrawal are recorded as unjudged, and a proposed correction that cannot be formed for the same reason is recorded as a failed proposal, separately from any effective correction.*

Proof. The verdict is a function of the claim, the played take, the script, and the rules at the revision of play, and the claim and script are available. If either of the other two cannot be recovered the function cannot be evaluated. The archive preserves the take, and the ledger preserves the revision's rules. □

The distinction between the admissible pool and the archive is the one drawn in Chapter 1. The pool decides what may be assigned, and the archive decides what may be adjudicated. A production that confuses the two will either keep serving a withdrawn take or lose the evidence needed to rule on reports about it.

7.7 What the Chapter Gives the Others

Chapter 8 begins from an upheld report and asks what rule it supports, how far its footprint reaches, and whether the correction can be formed. The tally above supplies its inputs: which of the stale classes a given audience reveals, which reports are violations and which are evidence for a rule, and the assets that must remain recoverable. Chapters 9 and 10 treat what the correction costs and whether the viewers can be moved. The three chapters share the pools and the audiences, so a single report can be followed through all of them.

The scripts `lawnmodel.py` and `verify_ch07.py` reproduce each figure of this chapter.

Chapter 8

Amendments, Footprints, and Correction Events

Chapter 7 ends with an upheld report. Upheld reports have two possible outcomes. A report may show that an edition violated a rule already in force, so that the assembly or the assigned take was at fault and the rules do not change. Or it may supply evidence for an amendment, so that the rules were incomplete and must be extended. Violations lead to repair, and this chapter is about the other outcome: how a rule is derived from one instance, how far it reaches, how the correction is made as a single event, when it cannot be made, and what a production can do when it cannot.

The chapter joins the earlier ones. Chapter 5 gave the footprint of a defect, Chapter 9 the capacity cost of an amendment, and Chapter 10 the repair of the viewers it invalidates. The correction event of this chapter is the unit that carries all three at one moment, so that the ledger never shows rules that the assignments do not meet. Production C is again the stress case, and it supplies the chapter’s last result: where replacement footage is shot matters for repair even when it makes no difference to capacity.

8.1 From One Instance to a Rule

An upheld report names a strip, a slot, and a stale class, and it is one instance of a defect. The rule it supports is a generalization of the instance, and there are three natural ones, which are the levels of Chapter 5: the interval of the defect, every showing of the entity, and every entity of the type. They cover different sets of stale classes. Table 8.1 gives, for each first report in each production, how many of the pool’s stale classes each level removes and the capacity that remains.

The table shows what a single instance can and cannot justify. The interval level removes exactly the stale classes that the instance represents, those of the same strip in the same persistence interval, and Proposition 5.4 of Chapter 5 proves it sound and complete for that interval. The type level removes every stale class of the pool, which is more than the

Prod.	First report	Stale classes removed			Capacity remaining		
	(strip, slot)	Interval	Entity	Type	Interval	Entity	Type
A (4 stale)	a, 2	2	2	4	351	351	238
	c, 4	1	1	4	422	422	238
	b, 5	1	1	4	404	404	238
B (7 stale)	a, 2	2	2	7	200	200	96
	b, 3	2	2	7	288	288	96
	c, 5	3	3	7	176	176	96
C (2 stale)	a, 2	1	1	2	1	1	1
	c, 5	1	1	2	2	2	1

Table 8.1: For each production and each first report, the number of the pool's stale classes removed by the interval, entity, and type levels, and the capacity that remains. The interval and entity levels coincide in every row because no stale class shows the reported strip in the other interval.

instance supports in every row: after the first report on strip a in production A, for instance, it removes two stale classes that the report does not represent. The type-level rule is justified only if the production has grounds to think the defect belongs to the whole type, and one instance does not supply them.

Definition 8.1 (Supported rule)**.** Given a set $\mathcal{R}$ of upheld reports, the *supported rule* is the union of the interval-level amendments of its reports. It can be raised to the entity level or the type level only on further evidence, namely reports of the same defect in other intervals of the entity or on other entities of the type.

The audience of production A reveals every pair of strip and interval that its pool contains, so the union of the interval amendments of its reports has the capacity of G on that pool, 238. That is a fact about this pool and audience, and the production could not have known it beforehand. Production C's audience reveals one of the two pairs in its pool, and the supported rule from its reports is the interval amendment on strip a alone, which leaves the defect on strip c in the pool for a later report.

8.2 The Order in Which Corrections Arrive

Reports arrive in an order that no one controls, and each upheld pair of strip and interval can trigger its own amendment. The order does not change where the production ends and it changes the path.

Proposition 8.2 (Order independence)**.** *Let amendments $X'_1, \dots, X'_n$ be applied in any order. The admissible set after all of them is $\bigcap_i X'_i$ whatever the order, and so is its capacity. The capacity along*

the way decreases from step to step, so the minimum capacity along any order is the final capacity, and the promise to an audience of U is kept at every step if and only if $\Game(\bigcap_i X'_i) \geq U$.

Proof. Removing realizations is commutative, since each amendment is a subset and applying one after another gives the intersection. Each removal can only reduce capacity, so the sequence of capacities is nonincreasing and ends at the capacity of the intersection. □

The intermediate values differ. For production A the three interval amendments, on strip a after its event, strip b after its event, and strip c before its event, can arrive in six orders. Every order ends at 238. The first step leaves 351, 404, or 422 depending on which strip is reported first, and the middle step leaves between 286 and 332. For production B the first step leaves 200, 288, or 176, the middle step between 128 and 168, and the end is 96. The paths are in Table 8.2. The differences matter to what follows, because the repair radii of Chapter 10 are computed against the pool at each step and not against the final pool.

Prod.	Order of reports	After first	After second	After third
A	a, b, c	351	286	238
A	a, c, b	351	303	238
A	b, a, c	404	286	238
A	b, c, a	404	332	238
A	c, a, b	422	303	238
A	c, b, a	422	332	238
B	a, b, c	200	168	96
B	a, c, b	200	128	96
B	b, a, c	288	168	96
B	b, c, a	288	144	96
B	c, a, b	176	128	96
B	c, b, a	176	144	96

Table 8.2: Capacity after each of three interval-level amendments, in every order. Premiere capacities are 494 for A and 320 for B.

8.3 The Correction Event

A correction is formed as one event, as the first chapter requires, with three parts at one effective time: the amendment of the rules, any footage added to the pool, and the repairs of the allocation. The reason for atomicity is that the ledger is the evidence of adjudication, and a ledger that shows the new rules beside the old assignments records editions that its own rules call inadmissible.

Definition 8.3 (Correction event)**.** A *correction event* at effective time τ is a triple of an amendment $X' \subseteq X$, an addition ΔT of takes to the pool, and a repair $a \mapsto a'$ of the allocation.

It is *formable* if the pool after the amendment and the addition admits an allocation for the audience, namely $\Cap(X' \cup \Delta X) \geq U$, and the repair keeps every moved viewer within the permitted budget.

The cost of not forming events atomically can be counted. Suppose the three amendments of production A are adopted in turn, each repaired afterward as in Chapter 10. If each rule enters the ledger before its repairs, the viewers whose editions the rule has just made inadmissible number 16, then 15, then 6, a total of 37 viewer-revisions on inadmissible editions. For production B the numbers are 8, 3, and 15, a total of 26. With atomic events the count is zero and the admissibility invariant of Chapter 1, that every assignment in the ledger meets the rules of its revision, holds after every event, which was checked for both productions.

8.3.1 When the Event Cannot Be Formed

An event can fail in two ways. The capacity condition can fail, as for production C under the agreement rule: the capacity is 1 for an audience of 2, and no repair of any radius exists. The budget condition can fail with capacity to spare. For A and B under the full agreement rule, the survivors-fixed repair needs radius 2, and with a budget of 1 no repair exists although 238 and 96 class tuples remain. The event of production A is formable at a budget of 2 or 3 and not at 1, and the same holds for B.

When an event cannot be formed there are four things the production can do: hold the amendment and keep the defect in the pool, which breaks the promise of Chapter 1 to the viewers who would see it; raise the budget, which disturbs viewers more; add footage; or relax a rule that the history permits. The last two are the subject of the next two sections.

8.4 Replacement Footage

Footage that is added to the pool restores capacity, and where it is added decides who can use it. For production C under G, cloning its admissible class at any one slot raises the capacity from 1 to 2, which is exactly the audience. The two class tuples that result differ at the cloned slot. The viewer who held the stale edition must move to the new tuple, and the distance depends on where the clone is. At slot 2, the slot of the stale class, the viewer moves one slot, from the stale class to its correct replacement, and the repair radius is 1. At every other slot the radius is 2: the viewer moves from the stale class at slot 2 and from the original class to the clone at the other slot. Table 8.3 gives all five.

Capacity alone cannot choose between the five placements, and the repair radius can. The footage that fixes a defect in place is the footage that costs the viewer least, which is a reason to reshoot the slot that held the error before shooting anywhere else. This is a statement about a single instance of C, and it need not hold in general, since a clone at a slot of heavy traffic can add many more tuples, as Chapter 9 showed for production A.

Clone at slot	1	2	3	4	5
Capacity	2	2	2	2	2
Repair radius	2	1	2	2	2

Table 8.3: Production C after *G*: the effect of adding one clone of the admissible class at each slot, for the two viewers of the premiere.

8.5 Relaxing a Rule That the History Permits

A rule can be relaxed when the story world admits the depiction it forbids. The direction rule forbids a cut between shots with opposite screen directions, and a story in which the character turns, or crosses the line of action on camera, permits the cut. Relaxing the rule at one edge enlarges the admissible set. Table 8.4 gives the capacity after *G* when the direction rule is dropped at each of the four edges of the scene.

Production	After *G*	Edge (1,2)	Edge (2,3)	Edge (3,4)	Edge (4,5)
A	238	310	364	270	260
B	96	200	200	192	192
C	1	2	1	1	1

Table 8.4: Capacity after the agreement rule when the direction rule is relaxed at one edge.

For production C, relaxing the first edge alone restores a second edition: slot 1 may be either of its two classes, since the rule no longer links it to slot 2. The repair is not cheap. The viewer who held the stale edition moves from the stale class at slot 2 to the correct class and from the first slot's class to the other, a radius of 2.

The model cannot say which edges the history permits. That belongs to the verdict of Chapter 7: the third step compares the verified appearance with the script history, and the same history decides whether a relaxation is licensed. A relaxation that the history does not license is not a correction but an error of a different kind, since it admits editions that depict a different story.

8.6 One Process

Footage, relaxation, the level of the rule, the budget, and the order of reports are parts of one decision. For production C after the first defect report, doing nothing keeps an edition that the viewer's report has shown to be wrong. The agreement rule forbids it and leaves the promise unkept. A clone at slot 2 keeps the promise at radius 1. A clone elsewhere keeps it at radius 2. A relaxation at the first edge keeps it at radius 2, at no storage cost and at the cost of a change to the film's grammar. None of these is forced by the mathematics. The mathematics says which are feasible and what each costs, in tuples, in disturbance, and in

footage.

The scripts `lawnmodel.py` and `verify_ch08.py` reproduce each figure of this chapter.

Chapter 9

Capacity and Robustness

The question that the book carries is how much variation a production can sustain while remaining open to correction. It has two precise answers. The first concerns capacity: after a correction, does the production still have enough distinct editions to serve its audience from scratch? The second concerns repairability: can the viewers reach those editions from the editions they hold now, within the disturbance that is permitted? This chapter gives the first answer and the next chapter gives the second. The two answers are different, and the chapter ends by showing why the first cannot stand in for the second.

The result that gives the chapter its shape is about the extreme case. A production with no spare capacity at all, one in which every admissible perceptual class is assigned to a viewer, can survive an amendment. It survives precisely when every class retains at least one admissible realization. A correction can therefore remove takes, even most of the takes, without removing an edition class, and a viewer whose assigned take is withdrawn can be given another take of the same class without any perceptible change. The failure case is the one the first chapter found in production C, where a valid correction eliminates an entire class and leaves one class for two viewers. The same test, applied to a perceptual threshold instead of exact equivalence, asks whether the corrected pool still admits a separated set large enough for the audience.

The chapter treats an amendment as a specified change to the set of admissible editions. Where the specification comes from, which constraints a continuity defect implicates and which shots it touches, is the subject of Chapter 5. Here the specification is given, and the question is what it costs.

9.1 Capacity and Spare Capacity

The setting is that of the first chapter. A production has a pool T_j of takes at each of m slots, and a map sending each take to its perceptual class. The *realizations* are the take tuples that the rules in force admit, and we write $X \subseteq \prod_j T_j$ for the set of them. The *class map* π sends a realization $\tau = (\tau_1, \dots, \tau_m)$ to its class tuple $\pi(\tau) = ([\tau_1], \dots, [\tau_m])$. Under the assumption of the first chapter that two editions are perceptually equivalent exactly when they have

the same class tuple, the number of perceptually distinct editions that the production can supply is the following.

Definition 9.1 (Capacity)**.** The *capacity* of a set X of realizations is $\mathrm{Cap}(X) = |\pi(X)|$, the number of class tuples that have at least one realization in X.

An *allocation* for an audience of U viewers is an assignment of a realization in X to each viewer such that no two assigned realizations have the same class tuple. By the pigeonhole argument of the first chapter, an allocation exists if and only if $U \leq \mathrm{Cap}(X)$. The *spare capacity* is $\mathrm{Cap}(X) - U$, and a production has *zero spare capacity* when every class tuple that X realizes is assigned to some viewer.

For the lawn scene of the first chapter the premiere capacities and audiences are those of Table 9.1. Production C has two class tuples and two viewers, so it has none to spare. Productions A and B have a great deal.

Production	Capacity	Audience	Spare capacity	Take tuples
A	494	60	434	120,042
B	320	30	290	10,240
C	2	2	0	204,800,000

Table 9.1: The lawn scene at the premiere, under the direction rule alone. These are the figures of the first chapter.

9.2 Amendments as Specified Changes

An amendment in this chapter is a change that removes realizations from the admissible set and is specified in advance.

Definition 9.2 (Amendment)**.** An *amendment* to a set X of realizations is a subset $X' \subseteq X$, the realizations that survive it. Its *loss* is $\mathrm{loss}(X') = \mathrm{Cap}(X) - \mathrm{Cap}(X')$, the number of class tuples that lose every realization.

Amendments that add footage are treated in Section 9.6. Three kinds of removal arise in practice, and they behave differently.

A *class-level* amendment removes every take of a class, usually because a rule excludes the class. The agreement rule of the first chapter is of this kind: it removes the takes that misdepict a strip. A *take-level* amendment removes particular takes by name, for instance because the rights to a shot have been withdrawn, because a file is corrupt, or because an asset has been excluded by digest. A *tuple-level* amendment forbids a combination of takes, for instance two takes that share a sound asset that may not be used twice, without removing either take from every combination.

For the lawn scene the amendments of Table 9.3 are used here and in the next chapter. Each stale class misdepicts one strip, and Table 9.2 records which. The agreement rule $\alpha(S, J)$

for a set of strips S and a set of slots J removes every class at a slot in J that misdepicts a strip in S. The slot sets are given and not derived here, and Chapter 5 shows how a defect determines them.

Production	Stale classes at slot j (screen direction, strip misdepicted)	Count
A	2 (L, a); 3 (R, a); 4 (R, c); 5 (L, b)	1 each
B	2 (R, a); 3 (L, b); 5 (R, c)	2, 2, 3
C	2 (L, a); 5 (R, c)	1 each

Table 9.2: The stale classes of the first chapter, with the strip that each one shows in the wrong appearance.

Name	Kind	Specification
α_a	class-level	$\alpha(\{a\}, \{2,3,5\})$, strip a at the slots where it must appear mown
α_b	class-level	$\alpha(\{b\}, \{3,5\})$
α_c	class-level	$\alpha(\{c\}, \{1,4,5\})$, strip c at every slot that shows it
G	class-level	$\alpha(\{a,b,c\}, \{1,\ldots,5\})$, the agreement rule of the first chapter
ω	take-level	withdraw one take from every class at every slot
x^*	class-level	withdraw every take of the single class that lies on the most class tuples

Table 9.3: The amendments used as running examples.

The capacities after each amendment are in Table 9.4. The amendment ω withdraws a take from each class and loses no class in any production, because every class has at least two takes: three in A, two in B, and forty in C. The take tuples fall from 120,042 to 15,808 in A, from 10,240 to 320 in B, and from 204,800,000 to 180,448,398 in C.

Production	Premiere	α_a	α_b	α_c	G	ω
A	494	351	404	422	238	494
B	320	200	288	176	96	320
C	2	1	2	2	1	2

Table 9.4: Capacity of each production after each amendment of Table 9.3.

Production C shows that touching a class and losing an edition are different things. The amendment α_b touches nothing in C, because C has no class that misdepicts strip b. The amendment α_c removes C's stale class at slot 5, a right-facing class that no direction-compatible edition of C could use, so it lay on no class tuple and capacity stays at 2. Only α_a

and G remove the stale class at slot 2, which lies on one of C's two class tuples.

9.3 Survival at Zero Spare Capacity

Theorem 9.3 (Survival of an amendment)**.** *Let $X' \subseteq X$ be an amendment and let $U \leq \mathrm{Cap}(X)$ be the audience. The promise of pairwise inequivalent editions can be kept after the amendment if and only if $\mathrm{Cap}(X') \geq U$. In particular, if the production has zero spare capacity, so that $U = \mathrm{Cap}(X)$, it survives the amendment if and only if $\pi(X') = \pi(X)$, which is to say that every class tuple retains at least one admissible realization. In that case every viewer can keep the class tuple the viewer already holds, and a viewer whose assigned realization was removed receives another realization of the same class tuple.*

Proof. An allocation in X' exists if and only if $U \leq \mathrm{Cap}(X')$ by the pigeonhole argument. If $U = \mathrm{Cap}(X)$ then $\mathrm{Cap}(X') \geq \mathrm{Cap}(X)$ forces $|\pi(X')| = |\pi(X)|$, and since $\pi(X') \subseteq \pi(X)$ this means $\pi(X') = \pi(X)$. Conversely, if $\pi(X') = \pi(X)$ then each viewer's class tuple has a realization in X', and assigning one realization of that class tuple to each viewer gives an allocation. Viewers whose assigned realization lies in X' keep it, and the others receive a different realization of the class tuple they held. □

The theorem separates two things that a correction can do. It can remove realizations, and it can remove classes. A production with zero spare capacity cannot afford to lose a class, but it can lose realizations without limit, short of losing the last realization of a class. The viewer whose take was withdrawn is not disturbed perceptually, because the replacement take is in the same class, and the ledger of the first chapter records the change of file without recording any change of edition.

Corollary 9.4 (Take-level amendments)**.** *Suppose the admissible set is defined by class-level rules, so that $X = \pi^{-1}(\pi(X))$, and let a take-level amendment withdraw a set of takes. A class tuple of $\pi(X)$ survives if and only if each of its slot classes retains at least one take. Consequently $\mathrm{Cap}(X')$ is the number of admissible class tuples all of whose slot classes retain a take, and a take-level amendment is free for a production with zero spare capacity if and only if it exhausts no class that lies on an admissible class tuple.*

Proof. The realizations of a class tuple κ are the products of the takes of its slot classes. After the withdrawal this product is nonempty if and only if every factor is nonempty. The rest follows from Theorem 9.3. □

A tuple-level amendment is the intermediate case. It removes some realizations of a class tuple and spares the others, and the class tuple survives if and only if at least one of its realizations is not forbidden.

Example 9.5 (Production C)**.** Production C has two class tuples and two viewers, so its spare capacity is zero. The amendment ω withdraws one take from each of its classes and leaves $39^5 \cdot 2 = 180{,}448{,}398$ take tuples and both class tuples, so by Theorem 9.3 the production

survives it and no viewer perceives a change. The amendments α_b and α_c also leave both class tuples. The amendments α_a and G remove the stale class at slot 2, which lies on one of the two class tuples, and the production is left with one class tuple for two viewers. The promise cannot be kept, although the production holds more than a hundred million take tuples. This is the failure case of the first chapter, and it is now an instance of a general test.

9.3.1 How much is lost

The loss of a class-level amendment is a count that the methods of Chapter 4 compute.

Proposition 9.6 (Loss from exhausted classes)**.** *Let an amendment exhaust a set W of classes, and suppose the admissible set is defined by class-level rules. Then* $\mathrm{loss}(X')$ *is the number of admissible class tuples that pass through at least one class of W. For a single exhausted class k at slot j this number is the count obtained by running the frontier method on the structure in which the set of classes at slot j is restricted to $\{k\}$. All these numbers, for every class k at every slot j, come from the forward tables T_{j-1} and the backward tables C_j of Chapter 4 at the cost of building them, together with one pass over the classes.*

Proof. A class tuple survives if and only if none of its slot classes is exhausted, so those that do not survive are exactly those through W. For the single class, the class tuples through k at slot j are the admissible class tuples of the restricted structure. For the last statement, the number through k at slot j is the sum, over assignments z of the frontier at $j-1$ that satisfy the constraints with largest slot j together with k, of $T_{j-1}(z)\,C_j(z')$, where z' is the resulting frontier at j, and this is a single pass over the entries of the tables. □

The class that lies on the most admissible class tuples is the one whose withdrawal costs the most, and it is the amendment x^* of Table 9.3. In production A it is the neutral class at slot 2, which lies on 260 of the 494 class tuples, because a neutral take is compatible with any neighbor. In B it is a right-facing class at slot 3 with 120 of 320, and in C it is the left-facing class at slot 1, which lies on both class tuples, so that withdrawing it leaves C with none. Each of these figures was checked by recomputing the capacity after the withdrawal.

9.4 Survival Under a Separation Threshold

If the promise is that editions differ by more than a tolerance ε under a pseudometric d on the realizations, the test is different. Assume as in the first chapter that realizations of the same class tuple are at distance 0. By the triangle inequality the distance between two realizations then depends only on their class tuples. Write $M_\varepsilon(X)$ for the size of a largest subset of X whose members are pairwise more than ε apart, the packing number of the first chapter.

Proposition 9.7 (Survival under separation)**.** *Under the hypothesis above, an allocation with pairwise dissimilarity greater than ε exists in X' for an audience of U if and only if $M_\varepsilon(X') \geq U$,*

and $M_\varepsilon(X') \le M_\varepsilon(X)$. If $U = M_\varepsilon(X)$, so that the audience exactly fills a largest separated set, the production survives the amendment if and only if the class tuples retained in $\pi(X')$ contain a set of U class tuples that are pairwise more than ε apart. Retention of every class tuple is sufficient for this and is not necessary.

Proof. The first two statements restate the definition of the packing number and the fact that a separated subset of X' is a separated subset of X. For the third, because distances depend only on class tuples, a separated set of realizations in X' corresponds to a separated set of class tuples in $\pi(X')$ with one realization chosen for each. If every class tuple of $\pi(X)$ is retained, a largest separated set of class tuples of $\pi(X)$ is retained. For the failure of necessity, a largest separated set can lie entirely in a proper subset of the classes. □

The toy example of the first chapter shows the failure of necessity. Four class tuples lie on a line at positions 0, 0.6, 1.2, and 1.8, the dissimilarity is the distance, and $\varepsilon = 1$. A largest separated set has two members, for example $\{0, 1.2\}$. Withdrawing any single class tuple leaves a separated pair, so the production at $U = 2$ survives each of the four withdrawals although it has lost a class tuple each time. It does not survive every withdrawal of two class tuples: withdrawing $\{0, 0.6\}$ or $\{0, 1.8\}$ or $\{1.2, 1.8\}$ leaves a packing of one.

The lawn scene gives a computed instance. Take the dissimilarity between class tuples to be the number of slots at which their classes differ, which is a pseudometric under which realizations of one class tuple are at distance zero, and take $\varepsilon = 1$, so that two editions count as separated if they differ at two or more slots. Production B at $\varepsilon = 1$ has $M_1 = 64$ at the premiere. With an audience of 64 it has zero spare capacity in the sense of this section. The results are in Table 9.5. The amendment α_b removes 32 of the 320 class tuples and leaves a separated set of 64, so B survives it, while α_a, α_c, and G leave 40, 40, and 32. A separated set of the full size survives the loss of classes that it did not use.

Production	ε	Premiere	α_a	α_b	α_c	G
A	1	105	75	94	87	54
A	2	29	21	28	23	16
B	1	64	40	64	40	32
B	2	20	16	18	16	12

Table 9.5: The packing number M_ε of the admissible class tuples under the Hamming dissimilarity, at the premiere and after each amendment.

Computing the packing number is a maximum independent set problem on the graph whose vertices are the class tuples and whose edges join tuples at distance at most ε, which is NP-hard in general [15]. The figures above were obtained by integer programming on instances of a few hundred vertices. For larger scenes only bounds are available, for instance a greedy separated set from below.

9.5 Robust Capacity

An anticipated family of amendments is a set of subsets of X. The question of how much a production can take is then a question about the worst member.

Definition 9.8 (Robust capacity)**.** Let $\mathfrak{A} = \{X'_1, \dots, X'_n\}$ be a family of amendments to X, regarded as alternatives, any one of which may occur. The *robust capacity* of the production for $\mathfrak{A}$ is $\mathrm{Rob}(\mathfrak{A}) = \min_i \mathrm{Cap}(X'_i)$, and the production is *robust to* $\mathfrak{A}$ *at audience* U if $U \le \mathrm{Rob}(\mathfrak{A})$. The *invariant capacity* is $\mathrm{Inv}(\mathfrak{A}) = \mathrm{Cap}\big(\bigcap_i X'_i\big)$.

The two quantities answer different questions, and the definition keeps them apart. Robustness allows each amendment to have its own allocation.

Proposition 9.9 (Robustness and invariance)**.** *The production is robust to* $\mathfrak{A}$ *at* U *if and only if for each* i *there is an allocation* a_i *in* X'_i*, and the allocations* a_i *may differ from one another. A single allocation that lies in every* X'_i *exists if and only if* $U \le \mathrm{Inv}(\mathfrak{A})$*. Moreover* $\mathrm{Inv}(\mathfrak{A}) \le \mathrm{Rob}(\mathfrak{A}) \le \mathrm{Cap}(X)$*, and* $\mathrm{Inv}(\mathfrak{A})$ *is also the capacity after all the amendments have been applied in turn.*

Proof. The first two statements apply Theorem 9.3 to each X'_i and to the intersection. The inequalities follow from $\bigcap_i X'_i \subseteq X'_i \subseteq X$ and monotonicity of capacity. Applying removal amendments one after another leaves the intersection of the survivor sets. □

The gap between the two can be large. For the family $\mathfrak{A} = \{\alpha_a, \alpha_b, \alpha_c, \omega\}$ the figures are in Table 9.6. Production A is robust at audience 60, since its robust capacity is 351, and it also keeps 238 class tuples after all four amendments have been applied in turn. Production B is robust at 30 with robust capacity 176 and invariant capacity 96. Production C is not robust at its audience of 2, since α_a leaves it with 1, and its invariant capacity is also 1. If the worst-class withdrawal x^* is added to the family, robust and invariant capacities become 234 and 108 for A, 176 and 72 for B, and 0 and 0 for C.

Production	Audience	Premiere capacity	Robust capacity	Invariant capacity
A	60	494	351	238
B	30	320	176	96
C	2	2	1	1

Table 9.6: Robust and invariant capacity for $\mathfrak{A} = \{\alpha_a, \alpha_b, \alpha_c, \omega\}$.

A production's required reserve for a family follows. It must hold enough spare capacity at the premiere to absorb the largest loss in the family, $\max_i \mathrm{loss}(X'_i)$, and if the amendments may all be applied in turn, the loss of the intersection. The family above has losses of at most 143 for A, 144 for B, and 1 for C, and the cumulative losses are 256, 224, and 1. The spare capacities of 434, 290, and 0 cover the first two and not the last.

9.6 Replacement Footage as Reserve

A correction can also add takes. A production that finds itself short of capacity can shoot more, and the amount of capacity that new footage buys depends on where it goes.

Proposition 9.10 (Cloning a class)**.** *Suppose the admissible set is defined by class-level rules and a production adds to slot j a new perceptual class that has the same screen direction and displayed appearance as an existing class k, so that it satisfies exactly the same constraints. The capacity increases by the number of admissible class tuples through k at slot j.*

Proof. The new class tuples are those that use the new class at slot j. By the hypothesis, such a tuple is admissible if and only if the tuple obtained by replacing the new class with k is admissible, so they are in bijection with the admissible class tuples through k at slot j, and none of them was counted before. □

The quantity is the same one that measured the cost of withdrawing the class. Footage in the class that lies on the most admissible tuples is the footage whose loss hurts most and whose duplication helps most. After the agreement rule, production C has capacity 1 and each of its class tuples passes through every class that remains. Adding one perceptually distinct clone of its admissible class at slot 5 raises the capacity to 2, which is exactly enough for its audience. In production A after G, a clone of the neutral class at slot 2 adds 130 class tuples, and in B a clone of the first class at slot 1 adds 24. Each increase was checked by recomputation.

The proposition gives a unit in which reserve footage can be priced. If a family of amendments requires a reserve of s class tuples, a production can buy it by shooting variants of the classes that lie on the most tuples, and by the proposition the number of variants needed is at most $\lceil s/m \rceil$, where m is the smallest number of tuples through the classes chosen, rounded up because a fraction of a variant cannot be shot. A qualification applies. A clone inherits every constraint of its source, so it can be vulnerable to the same amendment as the class it copies, and the capacity it adds counts as reserve only in the scenario where the amendment leaves it admissible. Useful reserve is the capacity that the footage adds in the scenario where it is needed. The minimal footage for a given reserve is a covering problem that this chapter does not solve.

9.7 Capacity Is Not Repairability

All the statements of this chapter are about the existence of an allocation. They say that, after an amendment, some assignment of realizations to viewers has distinct classes. They do not say that the viewers can be moved to that assignment from where they are. Production A after α_a has capacity 351 for an audience of 60, and a spare capacity of 291, and yet a viewer whose edition is invalid may find that the nearby admissible editions are all held by others. Whether the reserve is usable from a particular allocation depends on how the viewers are distributed among the classes and on how far each is permitted to move. That is the subject

of the next chapter, and the two chapters together give the book's central question its two answers: the capacity that a production keeps under correction, and the repairability of the allocation that it starts from.

The accompanying scripts `lawnmodel.py` and `verify_ch09.py` reproduce every figure of this chapter, including the exhaustive checks of capacity after each amendment, the loss of the worst class, the packing numbers by integer programming, and the effect of cloning a class.

Chapter 10

Allocation Under Change

Chapter 9 asked whether a corrected production still has enough distinct editions for its audience, and it counted editions as if the audience could be reassigned from nothing. A real audience is not reassigned from nothing. Each viewer holds an edition, a correction invalidates some of those editions, and the production has to move the affected viewers to valid ones while disturbing as few viewers, and as little, as it can. This chapter asks whether that can be done. It takes the pools of the previous chapter and the same three audiences, and it gives a second result that the first does not imply.

The result is that capacity is necessary and not sufficient. A pool can hold hundreds of unused class tuples and still fail to repair a viewer, because the unused tuples are not near enough to the viewer's edition and the nearby ones are held by other viewers. Section 10.3 exhibits such a viewer in the lawn scene with a spare capacity of 291. Repairability therefore depends on the allocation that the production starts from and on the disturbance that it is allowed to cause, and the chapter gives the exact condition and two simpler sufficient ones.

10.1 Allocations, Disturbance, and Budgets

The class tuples of the admissible pool before an amendment form a set P, and after it a subset $P' \subseteq P$, the set $\pi(X')$ of the previous chapter. An allocation is an injective map a from viewers to class tuples. Disturbance is measured by a pseudometric d on class tuples, which this chapter takes to be the Hamming distance, the number of slots at which two class tuples differ. A viewer moved from $a(v)$ to $a'(v)$ is disturbed by $d(a(v), a'(v))$, so a viewer who is given another take of the same class tuple is not disturbed at all.

Before the amendment each viewer holds a tuple of P. After it, the *invalid* viewers are those whose tuples lie outside P' and the *survivors* are the others. A *repair* is an allocation a' into P'. Three problems arise, and they differ in what may move.

Definition 10.1 (Repair problems)**.** Given a and P', the *survivors-fixed* problem asks for a repair a' with $a'(v) = a(v)$ for every survivor v. The *free* problem allows every viewer to move. Each comes in two forms: with a *per-viewer budget* r, the requirement is $d(a(v), a'(v)) \leq r$

for every moved viewer, and with a *total budget*, the requirement is that $\sum_v d(a(v), a'(v))$ be minimal or at most a stated bound. The *repair radius* r^* is the least per-viewer budget for which a repair exists, and r^*_{fix} and r^*_{free} denote it for the two problems, with $r^*_{\text{free}} \le r^*_{\text{fix}}$.

The inequality holds because any survivors-fixed repair is a free repair. The two radii can differ, and Section 10.3 gives a case.

10.2 The Exact Condition

For a per-viewer budget the problem is a matching problem. Let $B_r(t)$ denote the set of tuples of P' within distance r of t.

Theorem 10.2 (Repair with a per-viewer budget)**.** *A survivors-fixed repair with budget r exists if and only if the invalid viewers can be matched to distinct tuples of $F = P' \setminus \{a(v) : v \text{ a survivor}\}$ with each viewer matched inside its ball $B_r(a(v))$. By Hall's theorem [4] this holds if and only if for every set W of invalid viewers*

$$\Big| F \cap \bigcup_{v \in W} B_r\big(a(v)\big) \Big| \;\ge\; |W|.$$

A free repair with budget r exists if and only if the same condition holds with all viewers in place of the invalid viewers and P' in place of F. Either condition can be decided in time polynomial in the size of the candidate graph by maximum bipartite matching. That graph is given explicitly here, and it can itself be large relative to the stored footage. For a positive separation threshold, replacements that are distinct can still conflict with one another, so ordinary matching no longer settles the problem.

Proof. A survivors-fixed repair gives each invalid viewer a tuple of P' that no survivor holds and no other invalid viewer holds, which is a matching into F within the balls, and conversely. The Hall condition is the standard characterization of the existence of such a matching. The free case is the same argument with every viewer required to receive a tuple, a survivor being allowed to keep its own because $a(v) \in B_r(a(v))$ whenever $a(v) \in P'$. □

A total budget is a minimum-cost assignment. The minimum total disturbance of a survivors-fixed repair is the optimum of the assignment problem between the invalid viewers and F with cost $d(a(v), t)$, and for the free problem it is the assignment of all viewers into P', both solved in polynomial time by the Hungarian method [16]. A common objective adds a fixed cost λ for each viewer who moves at all, so that a repair that disturbs few viewers a little is preferred to one that disturbs many.

10.2.1 Why Capacity Does Not Settle It

Proposition 10.3 (Capacity is necessary, not sufficient)**.** *If a repair exists then $|P'| \ge U$. The converse fails: there are allocations with $|P'| \ge U$ and no repair of a given budget r.*

Proof. A repair is an injective map of U viewers into P'. For the failure, take an invalid viewer all of whose valid tuples within distance r are held by survivors. No survivors-fixed repair with budget r places that viewer, whatever the size of P'. Section 10.3 exhibits such a viewer with $r = 1$. □

The abstract form of the example in the outline is this. Suppose a pool holds a hundred distinguishable editions and the audience is sixty. A correction removes the editions of two viewers, and each of the two has the same sole acceptable replacement within the permitted disturbance, a single free tuple that lies in both balls. A hundred editions exist, and forty of them are unassigned, but the Hall condition fails for the set of the two viewers, because their balls hold one free tuple between them. Capacity asks whether sixty distinct editions exist, and repairability asks whether these two viewers can reach two.

10.3 The Shared Example

The lawn scene supplies the audiences of Chapter 9: sixty viewers for production A, thirty for B and two for C, each drawn from the premiere pool by a fixed pseudorandom choice, standing in for the keyed permutation of the ledger. The amendments are those of Table 9.3 of the previous chapter. Table 10.1 gives for each production and amendment the number of invalid viewers, the capacity that remains, the repair radii, the minimum total disturbance of a survivors-fixed repair, and the smallest radius at which every invalid viewer’s ball holds at least U valid tuples.

Prod.	Amend.	Invalid	Capacity	r^*_{fix}	r^*_{free}	Total cost	Ball radius
A	α_a	16	351	1	1	16	3
A	α_b	15	404	1	1	15	3
A	α_c	6	422	1	1	6	3
A	G	31	238	2	2	37	4
B	α_a	8	200	1	1	8	3
B	α_b	3	288	1	1	3	3
B	α_c	15	176	1	1	15	3
B	G	20	96	2	2	27	4
C	α_a, G	1	1	none	none	none	none
C	α_b, α_c	0	2	0	0	0	0

Table 10.1: Repair of one audience per production after each amendment. The ball radius is the smallest r for which every invalid viewer’s ball B_r holds at least U valid tuples, a sufficient condition for repair.

For A and B the single amendments are repaired at radius 1 and the full agreement rule G at radius 2, and the total disturbance of the survivors-fixed repair is the number of invalid viewers plus a few extra slots: 37 slots for the 31 invalid viewers of A under G, and 27 for the

20 of B. For production C the audience of two sits at zero spare capacity, so Theorem 9.3 of the previous chapter decides it: α_b and α_c leave both class tuples and need no repair, while α_a and G remove one of the two, and no repair of any radius exists. This is the second form of the failure case, and it is a failure of capacity and not of distance.

10.3.1 Sufficient Conditions

Proposition 10.4 (The ball condition)**.** *If every invalid viewer's ball $B_r(a(v))$ contains at least U tuples of P', then a survivors-fixed repair with budget r exists.*

Proof. Survivors hold at most $U - k$ tuples of P' when k viewers are invalid, so for any set W of invalid viewers the free tuples in the union of their balls number at least $U - (U - k) = k \geq |W|$. This is the Hall condition of Theorem 10.2. □

The condition is conservative. It guarantees a repair at radius 3 for the single amendments of Table 10.1 and at radius 4 for G, where repairs in fact succeed at 1 and 2. Its use is that it can be checked from the pool alone, by counting tuples in balls, without reference to the allocation of the other viewers.

10.3.2 A Viewer That Capacity Cannot Reach

Capacity and radius come apart in an actual allocation of the lawn scene. Take the audience of sixty for production A drawn with seed 6 in the search of Table 10.2, and apply α_a. Twelve viewers are invalid, 351 class tuples remain, and 303 of them are free, which is a spare capacity of 291 over the audience. One invalid viewer, holding a class tuple that the amendment has removed, has exactly two valid class tuples within Hamming distance 1, and both are held by survivors. The Hall condition fails for that single viewer at $r = 1$, so $r^*_{\text{fix}} = 2$. The free problem has $r^*_{\text{free}} = 1$: a survivor can step aside by one slot, and the invalid viewer takes the tuple it vacates.

Which of these the production should do is a policy matter. The first leaves every survivor exactly where the survivor was, and costs the invalid viewer a larger disturbance. The second disturbs one extra viewer by one slot to spare the invalid viewer a larger one, which the total-budget objective with a movement cost λ weighs. Table 10.2 gives the worst survivors-fixed radius found over four hundred random audiences for each case.

10.4 Sequences of Amendments

Corrections arrive one after another, and the repairs accumulate. Because the disturbance is a metric, the displacement of a viewer over a sequence of repairs is bounded by the sum of the individual displacements.

Proposition 10.5 (Drift)**.** *If a sequence of amendments is repaired in turn with per-viewer budgets $r_1, \ldots, r_n$, a viewer's final class tuple is within $\sum_i r_i$ of its original one. A single repair from the*

Prod.	Amendment	Capacity	Worst r^{*}_{fix} over 400 audiences
A	α_a	351	2
A	α_b	404	2
A	α_c	422	2
A	G	238	2
B	α_a	200	2
B	α_b	288	2
B	α_c	176	2
B	G	96	3

Table 10.2: The worst repair radius over four hundred random audiences, survivors fixed. Every row has far more capacity than audience, and every worst case needs radius at least 2.

original allocation directly into the pool that results from all n amendments has radius that need not equal $\sum_i r_i$, and can be smaller.

Proof. The first statement is the triangle inequality for d. For the second, Table 10.3 gives an instance in which successive radii sum to more than the direct radius. □

Applying α_a, α_b and α_c to production A in turn, each repair has radius 1, and the largest accumulated movement of any viewer is 2, with 31 viewers moved at least once, although each repair alone moved at most the viewers it found invalid. A single repair straight into the final pool has radius 2. For B the three repairs have radii 1, 1 and 2, the largest accumulated movement is 3, and the direct repair has radius 2. A production that repairs promptly keeps each step small, and a production that waits and repairs once can have a smaller largest disturbance, at the price of leaving viewers on invalid editions for longer. The choice belongs to the release policy of a later chapter.

Prod.	Radii	Capacity per step	Max drift	Moved	Direct
A	1, 1, 1	351, 286, 238	2	31	2
B	1, 1, 2	200, 168, 96	3	20	2

Table 10.3: Repair in turn under α_a, α_b, α_c for the audiences of Table 10.1, each step a minimum-cost survivors-fixed repair.

10.5 Reproducibility

A repair has to be a function of the ledger. The pool after each amendment is determined by the correction events recorded in it, and the allocation before the amendment is determined by the ledger’s assignment records. If the assignment problem is solved by a fixed algorithm

with a fixed order on class tuples for breaking ties, the repair is determined as well, so the same ledger yields the same repaired allocation on every replay. For the lawn scene, two replays of the minimum-cost repair of the keyed allocation, with a hash-sorted audience, gave identical assignments for production A under G with thirty invalid viewers and for B with twenty-three. Ties among equal-cost repairs are the only freedom, and the tie-breaking rule is part of the specification of the production.

10.6 What the Two Chapters Give Together

Chapter 9 gives the capacity that a production keeps and the robustness that a family of amendments leaves. This chapter gives the repairability of an allocation within that capacity. The first is a property of the corrected pool alone and the second a property of the pool together with where the viewers stand and how far each may be moved. Production C fails the first and so fails the second. Production A, after α_a, passes the first with a spare capacity of 291 and fails the second for one viewer at radius 1 in the allocation of Section 10.3. The remaining reserve question is how much of the capacity is usable from a particular allocation, which the ball condition bounds from below and the matching of Theorem 10.2 decides exactly.

The scripts `lawnmodel.py` and `verify_ch10.py` reproduce each figure of this chapter.

Part IV

Release and Repertoire

Chapter 11

Locales and the Worldwide Premiere

A release has one premiere instant and many local productions. Each has its own cast, language, and setting, and each serves its own audience. The promise of Chapter 1 was that no two viewers receive the same film, and the promise has a scope: within a production, which the earlier chapters have handled, or across the whole release. This chapter takes the worldwide scope. The criterion of uniqueness that Chapter 2 settled decides what composes. If uniqueness means exact inequivalence, local allocation composes when the local productions share no class. If it means separation by more than a tolerance, sharing no class is not enough, because editions selected independently in two locales must also be far enough apart across them.

The concrete case is a sequence made of shared footage, in which the local cast, language, and setting leave no perceptible trace. The chapter treats the shared specification of the story, the passages at which editions of different productions may coincide, and the minimum footage that a locale has to produce to serve its audience. Its results are computed on the lawn scene, with locales that use the pool of production A, and the footage question is also answered for an abstract pool so that its dependence on the audience and the tolerance can be seen.

11.1 One Specification, Many Productions

What is shared among the locales is the specification of the story: the story world and its events, the script history, the scene plan with its slots and times, and the rules in force. What is local is the footage. A *local production* supplies a pool of takes at each slot, shot with the locale's cast, in its language, in its setting, and the editions of a locale are drawn from its own pool.

A slot carries a *local trace* if a viewer who compared the locale's takes there with those of another locale could tell them apart on account of the cast, the language, or the setting. A slot with no local trace is *shared*: the same footage, or footage that differs imperceptibly, serves every locale. An establishing view of the lawn with no person in it is the standard case. Let ℓ be the number of slots at which every edition carries a local trace, and $s = m - \ell$

the number of shared slots. In the lawn scene the mower appears in slots 2, 3 and 4, so these carry a local trace and slots 1 and 5, which show only the lawn, are shared.

Whether a slot carries a local trace is a fact about footage and viewers. It has to be established for each production, and a production that assumes it will have counted its way to a promise that nobody checked.

11.2 Exact Inequivalence Across Locales

Chapter 1 gave the condition for worldwide uniqueness under exact inequivalence. Let G_p be the set of global classes that locale p can realize and U_p its number of viewers.

Proposition 11.1 (Worldwide feasibility, restated)**.** *Pairwise inequivalent editions for all viewers, each viewer receiving an edition of the viewer's own locale, exist if and only if for every set P of locales $|\bigcup_{p\in P} G_p| \geq \sum_{p\in P} U_p$. If the sets G_p are pairwise disjoint, this reduces to $U_p \leq |G_p|$ for each locale separately.*

This is Proposition 1.14. Its content for the release is that allocation composes by locale exactly when the class sets are disjoint, and it fails to compose otherwise. A shared sequence is the case in which it fails to compose.

Take two locales that both use the 238 class tuples of production A after the agreement rule, and suppose the sequence is fully shared, so that $\ell = 0$ and a class tuple in one locale is the same global class as the same class tuple in the other. The union of their global classes has 238 elements. Let the first locale have 150 viewers and the second 100. Each locale's audience fits within 238, so the local test passes in both, and the combined audience of 250 exceeds the 238 global classes. The Hall condition fails for the pair, and some viewers in the two locales must receive the same film. If a single slot carries a local trace the global classes become 476 and the allocation is feasible. The difference between infeasible and feasible is one slot of local footage.

11.3 Perceptual Separation Across Locales

If the promise is separation by more than ε under the Hamming dissimilarity of Chapter 2, disjointness of classes is not enough. Two editions of different locales that differ only at slots with a local trace are different, and the question is how different.

Proposition 11.2 (Cross-locale distance)**.** *Let every edition of every locale carry a local trace at ℓ slots, with local footage at a slot differing perceptibly between locales. Then any two editions of different locales differ at all ℓ of those slots, so their Hamming dissimilarity is at least ℓ, and it exceeds ℓ by the number of shared slots at which their classes differ. Consequently, for every tolerance $\varepsilon < \ell$, locale-by-locale allocation is sound: if each locale allocates editions that are pairwise more than ε apart within the locale, then all editions of the release are pairwise more than ε apart, and the packing number of the union is the sum of the local packing numbers.*

Proof. Editions of different locales use different footage at every local-trace slot, so they differ there, and the Hamming dissimilarity counts one for each such slot. Any pair within a locale is separated by assumption, and any cross pair has dissimilarity at least $\ell > \varepsilon$. □

For $\varepsilon \geq \ell$ the proposition says nothing, and the numbers show what happens. Take the lawn scene with two locales that use the pool of production A, with local traces at slots 2, 3 and 4, so that $\ell = 3$. Table 11.1 gives the packing numbers. At tolerances 0, 1, 2 and 3 the union packs to exactly the sum of the local numbers. At $\varepsilon = 3$ the proposition's guarantee has run out, and the sum happened to hold here because the two locales' separated sets can be chosen to differ at a shared slot. At $\varepsilon = 4$ it fails: each locale packs 3, the sum is 6, and the union packs only 4.

Local-trace slots ℓ	$\varepsilon = 0$	1	2	3	4
Local packing M_ε (each locale)	238	54	16	7	3
Sum of the two locales	476	108	32	14	6
Union, $\ell = 3$ (slots 2–4 local)	476	108	32	14	4
Union, $\ell = 1$ (slot 2 local)	476	98	26	7	3
Union, $\ell = 0$ (fully shared)	238	54	16	7	3

Table 11.1: Packing numbers of the editions of two locales that use the 238 class tuples of production A after the agreement rule, under Hamming dissimilarity, for three amounts of local trace.

The rows with fewer local slots show the same effect earlier. With one local slot the union packs to the sum at $\varepsilon = 0$ and falls short from $\varepsilon = 1$, 98 against 108. With none, the union packs to the single locale's number at every tolerance, since the two locales' editions coincide and the second locale adds nothing.

11.4 Passages Where Editions May Coincide

The local-trace slots decide where coincidence can happen. Two editions of different locales coincide, as global classes, only if they agree at every slot, and they cannot agree at a slot that carries a local trace. A whole edition can therefore coincide with an edition of another locale only if the sequence has no local trace at all. A film has many scenes, though, and uniqueness can rest on any of them. A scene that is cast-free and fully shared contributes variation within a locale, and contributes nothing to separation between locales, because every locale offers the same choices there.

Proposition 11.3 (Where cross-locale separation comes from)**.** *The Hamming dissimilarity between two editions of different locales is ℓ plus the number of shared slots at which their classes differ. Two locales that make the same choices at every shared slot are therefore separated across locales by exactly ℓ, and if $\ell = 0$ they can coincide. Shared slots can still contribute distance when the chosen*

editions differ there. What a local trace supplies is a guaranteed contribution of ℓ irrespective of the choices at shared slots. A second distinction concerns what is counted. The packing number of the union is the size of the largest set of editions of the release that are pairwise separated. A release needs more than that: it needs a packing that meets a quota for each locale. A large combined packing may assign too many of its positions to one locale and too few to another, so the union's packing number is an upper bound on the sum of what the locales can be served and not a statement that each locale's audience is served.

The design consequence is that a production that depends on a scene being shared across locales should not count on it for worldwide distinctness. The scenes that carry cast are the scenes whose variation separates locales, and the scenes that do not are the scenes that save footage. A release plans its footage by deciding which scenes are which.

11.5 The Footage a Locale Must Produce

The audience of a locale is given and the footage is the cost. For the exact promise, a locale needs enough class tuples for its audience. Without further rules, a pool with c_j classes at slot j has $\prod_j c_j$ class tuples, and the footage is $\sum_j c_j$ classes. Table 11.2 gives the least total number of classes over five slots that reaches an audience of U.

Audience U	Least total classes	Classes per slot
2	6	$1, 1, 1, 1, 2$
30	10	$2, 2, 2, 2, 2$
60	12	$2, 2, 2, 2, 4$
238	15	$3, 3, 3, 3, 3$
1,000	20	$4, 4, 4, 4, 4$

Table 11.2: Least total number of classes over five slots whose product reaches the audience, with no rule restricting the combinations.

The footage grows like the mth root of the audience. An audience of a thousand needs twenty classes, four at each slot, which is fifty times fewer classes than viewers. This is the economy the book's framing section relies on, and it holds for exact distinctness.

Tolerance changes the answer sharply. If editions must be more than ε apart under the Hamming dissimilarity, a pool with q classes at each of m slots has at most $q^{m-\varepsilon}$ pairwise separated class tuples, by the Singleton bound, since a set of tuples that differ in more than ε slots is determined by any $m - \varepsilon$ of its coordinates.

Proposition 11.4 (Footage for separation)**.** *To serve U viewers with editions pairwise more than ε apart, a uniform pool needs at least $q = \lceil U^{1/(m-\varepsilon)} \rceil$ classes at every slot. The bound is attained whenever there is a code of length m over an alphabet of q symbols with minimum distance $\varepsilon + 1$ and at least U words, for instance a Reed–Solomon code when q is a prime power with $q \geq m - 1$.*

Proof. Two separated tuples differ in at least $\varepsilon + 1$ coordinates, so their restrictions to any fixed $m - \varepsilon$ coordinates are distinct, and a set of U separated tuples has $U \le q^{m-\varepsilon}$. For the attainment, a Reed–Solomon code of length $m \le q + 1$ and dimension $m - \varepsilon$ has minimum distance $\varepsilon + 1$ and $q^{m-\varepsilon}$ words. □

For five slots and an audience of sixty the bound requires 3, 3, 4 and 8 classes per slot at tolerances 0, 1, 2 and 3, against a total of 12 classes in the best non-uniform pool at tolerance 0, and the explicit codes reach it: a parity code over three symbols with 81 words and minimum distance 2 at $\varepsilon = 1$, a Reed–Solomon code over a field of four elements with 64 words and distance 3 at $\varepsilon = 2$, and one over a field of eight elements with 64 words and distance 4 at $\varepsilon = 3$. All three were constructed and their distances checked. In total classes over five slots these are 15, 20 and 40, so each step in tolerance costs more footage than the one before. For an audience of 238 the lower bounds are 3, 4, 7 and 16, and their attainment was not checked.

Shared footage reduces what a locale must shoot. In the lawn scene let slots 1 and 5 be shared with two classes each, so that a locale inherits 4 combinations from them and shoots only at the three local slots. For an audience of 30, 60 and 238 the locale needs 2, 3 and 4 classes at each local slot, which is 6, 9 and 12 classes, against 10, 12 and 15 if all five slots were local. The saving is the footage of the shared slots, and it comes at the price of Proposition 11.3: the locale's distinctness from other locales rests on its three local slots.

11.6 What the Chapter Establishes and What It Assumes

The propositions hold for any locales, any pseudometric of the form used here, and any scene plan. What the figures depend on is stated beside them: that local footage at a local-trace slot differs perceptibly between locales, which is a fact about footage and viewers; that the Hamming dissimilarity on class tuples stands in for a measured dissimilarity, as in Chapter 2; and that the scene is a five-slot lawn whose cast appears in three slots. A release with a hundred scenes has a hundred such decisions, and the chapter's counts show how the footage and the audience respond to them, not what they are.

The next chapters take the editions of a release through time. Chapter 12 asks what a ledger must retain to recover an edition on a given date, and Chapter 13 asks how a viewer's finite life limits how many of these editions the viewer can ever see.

The scripts `lawnmodel.py` and `verify_ch11.py` reproduce each figure of this chapter.

Chapter 12

Ledgers, Archives, and the Release Slider

The preceding chapters allocate editions and correct them. All of it presupposes a record: the ledger that says which viewer was shown which edition, under which revision, and the archive of footage from which that edition can be put together again. A correction that withdraws a take does not erase what a viewer saw, and a report from a viewer is adjudicated against the edition that was played, so the record decides what can be adjudicated at all. This chapter treats what the ledger holds and how much it costs, what an archive has to retain to reconstruct a played edition, what a ledger of unique editions does to the privacy of the people in it, and what a release slider should show for each date, including a date before the premiere and a scheduled premiere to which no edition has yet been assigned.

All counts are taken on the lawn scene with the allocations of Chapter 10: sixty viewers of production A, thirty of B, and two of C, drawn once at random from the premiere pool, with one take per slot drawn at random for each viewer. They show what the record costs in this scene. They do not measure storage, since no byte count of any take exists in this book.

12.1 The Ledger and Its Records

A *record* is a viewer reference, a revision identifier, a class tuple, and a take tuple, together with the date of assignment. The revision identifier is what makes the other fields decodable: the same tuple of indices names different footage under different pools. A *ledger* is the sequence of records, appended and never overwritten, so that a viewer who has been repaired once has two records and the earlier one is still there.

The tuple fields are small. Table 12.1 gives their size under two encodings. The self-describing encoding stores for each slot the index of the class and the index of the take, with as many bits as the largest pool at that slot requires. The compact encoding stores a single index into the space of take tuples, which needs $\lceil \log_2 K \rceil$ bits for K take tuples, and can be decoded only with the pool of the named revision.

Production	Take tuples K	Compact bits	Per slot: class + take	Total
A	120,042	17	$13 + 10$	23
B	10,240	14	$14 + 5$	19
C	204,800,000	28	$5 + 30$	35

Table 12.1: Bits needed to record one edition at the premiere, by encoding. The viewer reference, the revision identifier, and any signature are not counted.

The length of the compact encoding is a different quantity from the numerator of the rate of Chapter 1. The rate counts perceptually distinguishable editions, $\log_2 K$ for K of them. The encoding names an assembly, and its length is $\lceil \log_2 T \rceil$ bits for T admissible take tuples. At the same revision the two can differ by the whole amount. After the lawn correction production C has 102,400,000 take tuples, so a ledger entry costs twenty-seven bits, against one perceptual edition and zero bits of distinguishable choice. The cost of recording the film is higher than the amount that the viewer can tell about it, and in this case by the whole amount.

12.2 What an Archive Retains

A record is useless without the footage it names. The pool of a production holds sixty-nine takes in A, fifty-eight in B, and four hundred in C. Some of them belong to classes that the corrections of Chapters 7 to 10 remove from the admissible pool: after the agreement rule, twelve of A’s takes, fourteen of B’s, and eighty of C’s are in stale classes, and the admissible pool shrinks to fifty-seven, forty-four, and three hundred twenty takes. Three retention policies are possible.

Definition 12.1 (Retention policies)**.** An *archive-all* policy keeps every take ever shot. An *admissible-pool* policy keeps only the takes of classes that are admissible under the current revision. A *referenced* policy keeps the admissible pool and, in addition, every superseded take that some record in the ledger names.

An edition can be reconstructed from a record only if every take it names is still held, so the policies differ in which viewers can be served. Table 12.2 gives the counts after the final correction.

Proposition 12.2 (Cost of reconstruction)**.** *Let R be the set of takes named by the records of played editions. If the system must retain the ability to serve every current admissible take and to reconstruct every played edition from its recorded take references, the least archive is the union of the admissible takes of ρ and R, which is the referenced policy. If the obligation is only to reconstruct played editions and to sustain a feasible current allocation, a smaller archive can suffice. Its size exceeds the admissible pool by the number of stale takes in R, and no more than the archive-all policy.*

	Shot	Admissible	Stale	Stale takes named by a record	Viewers holding a stale class	Not reconstructible if stale takes are purged
A	69	57	12	12	31 of 60	31
B	58	44	14	12	20 of 30	20
C	400	320	80	1	1 of 2	1

Table 12.2: Footage retained and viewers who can be reconstructed after the agreement rule, by policy. The last column is the number of viewers whose played edition names a take that an admissible-pool policy has discarded.

Proof. An edition is reconstructed from the takes it names, so every take of R is necessary. The admissible takes are necessary to serve the current revision, and nothing else is needed. □

In the lawn scene the referenced policy keeps 69 takes in A, which is all of them, since the sixty viewers between them name every one of the twelve stale takes. In B it keeps 56 of 58, and in C it keeps 321 of 400, because the one viewer who holds a stale class in the corrected production holds one take of the eighty. Archive-all would have kept the eighty. The saving is large in production C and nil in A, and the difference is the size of the audience relative to the stale pool: sixty viewers cover twelve stale takes, and two viewers cover one of eighty.

A take-level amendment sets the same trade. Withdrawing the first variant of every class in the premiere pool touches 96 of the 300 take slots held by the viewers of A, which is 56 of the 60 viewers, 72 of 150 in B, and 1 of 10 in C. The withdrawn takes cannot go back into any edition, and they have to stay in the archive for as long as a played edition names them.

The retention rule has a corollary that connects to the adjudication of Chapter 7. A report about a played edition is decided against the take that was shown. Where the take has been discarded, the report can be assigned only the verdict unsupported, since there is nothing to inspect. The archive therefore has to keep a take for at least as long as a report can still be filed against it, and the retention period is the report window.

12.3 Privacy in a Ledger of Unique Editions

The promise of this book is that each viewer receives a different film. A record that identifies the film therefore identifies the viewer, and the property that makes the cinema individual makes its ledger sensitive. The question is how much of an edition a third party has to know to single out its viewer.

Let a person who knows what a viewer saw at some set of s slots look for that viewer in the ledger. The *anonymity set* is the number of records consistent with the observation. Table 12.3 gives its mean over viewers and over the choices of observed slots, for the allocations above, at the class level.

Observed slots s	1	2	3	4	5
A (60 viewers)	15.95	4.96	2.12	1.29	1.00
B (30 viewers)	7.44	3.13	1.71	1.23	1.00
C (2 viewers)	1.80	1.60	1.40	1.20	1.00

Table 12.3: Mean number of viewers whose recorded class tuple agrees with a given viewer's at the s observed slots, averaged over viewers and over all choices of the slots.

Two things follow from the arithmetic. First, the anonymity set is one when all five slots are observed, for every production, since this is what uniqueness means. Second, it falls quickly: in A, a person who knows what a viewer saw at three of five slots is looking at about two candidates. The class tuple is a quasi-identifier, and in a population this small it is nearly a name.

Proposition 12.3 (The tuple is the identifier)**.** *If every viewer of a release has a distinct class tuple, then a ledger that stores tuples without viewer references still partitions the audience into singletons, and any observation of a viewer's edition that fixes the whole tuple singles out the record. Singling out a record is not identifying a person: linking the record to a civil identity requires information beyond the tuple, such as the viewer reference or the receipt. Pseudonymous viewer references protect the viewer only against a party who has no observation of the edition and no means of linking one to the other. The anonymity sets of the table show how quickly the candidate set shrinks under observation, and they do not show that any such linkage exists.*

This is a statement about what the design commits to and not about any attack. It has three consequences for the ledger. The mapping from a viewer to a tuple is the sensitive object, so it should be held apart from the archive and the adjudication, which need only the tuple. The viewer can hold a receipt, so that a report can be filed from the receipt alone without the ledger linking it to an account. And the ledger should record only as much of an edition as recovery and adjudication need, which for the slotwise encoding is the whole tuple, and for a slot-local complaint can be a single slot. None of this bounds what a viewer who publishes a receipt discloses about themselves, and the shared-viewing exchange of Chapter 1 depends on viewers doing just that.

12.4 The Release Slider

The *release slider* is the interface by which a viewer asks to see the film as of a date. It has to be defined for every date, because a viewer can move it anywhere. For a viewer v with records $r_1, \dots, r_n$ of dates $d_1 < \cdots < d_n$, and a date d, the slider shows the edition of the last record dated at or before d, under that record's own revision. This rule gives a definite answer on each of three kinds of date.

For a date before the premiere, no record has been dated at or before d, and the slider shows no edition. It can still show the schedule and the fact that an assignment has not

been made. It must not show a preview of the edition the viewer will receive, because the premiere pool at that date may not be the pool at the premiere, and a preview would be an assignment under another name.

For a scheduled premiere with no edition yet, the slider shows the booking and a sentence saying that the edition is assigned at the premiere. Whether to assign at booking or at the premiere is a design choice with a measurable difference. If assignments are made at booking and an amendment arrives before the premiere, the booked assignments that touch removed classes are invalid and have to be repaired in the ledger. In the lawn scene the first amendment alone invalidates 16 of the sixty booked assignments in A and 8 of thirty in B, repaired at radius one. The viewers have seen nothing, so no change is visible, and the records are nevertheless rewritten. If assignment is deferred to the premiere, the viewers receive editions from the corrected pool and no repair is needed. Deferral costs the production the ability to confirm early that its audience fits, and fitting is a capacity question (Chapter 9) that can be asked of the booking count without assigning anything.

For a date after the premiere, the slider shows the record in force. This is where corrections show. Table 12.4 follows a viewer population through the three amendments of Chapter 8 applied in sequence, with the repair of Chapter 10 at each step: the viewers who held an invalid tuple move to the nearest free valid one, and the others keep what they had.

	Pool, α_a	α_b	α_c	Moved at each step	Viewers ever moved	Cost, sequence / one shot
A	351	286	238	16, 15, 6	31	37 / 37
B	200	168	96	8, 3, 15	20	28 / 27
C	1	1	1	no repair possible	—	—

Table 12.4: The slider's history for the sixty, thirty, and two viewers of A, B, and C under the three amendments in sequence, with each repair at minimum total Hamming movement, survivors fixed. Moves are counted as new records.

In A the sequence moves 37 records for 31 viewers, which is the same as the viewer-level revisions of Chapter 8, and no viewer's slider changes more than twice. The total movement of the sequence equals the one-shot repair of the final pool in A, and exceeds it in B by one, 28 against 27. The reason is that a viewer moved at the second step can land in a place the third step removes, so a viewer in the sequence may be moved more than once. Production C has a pool of one after the first amendment and an audience of two. No sequence of repairs can give both viewers distinct tuples, and the slider for C shows, after the first amendment, either one edition for two viewers or a viewer without one. The ledger records the failure, and it is a failure of capacity and not of bookkeeping.

The slider also shows what anticipated allocation buys. If the premiere assignments are drawn from the pool that survives every amendment, the invariant pool of Chapter 9, A has

238 available for 60 viewers and B has 96 for 30. No viewer is ever moved, the slider is flat, and the ledger has one record per viewer. C has an invariant pool of one for two viewers, so anticipation is impossible there. The price of anticipation is that the premiere offers only 238 of the 494 class tuples of A, which is the capacity given up.

12.5 What the Chapter Establishes and What It Assumes

The propositions on retention and identification hold for any ledger of the kind defined here. The numbers rest on stated assumptions. The take counts stand in for storage, and no storage cost has been measured. The viewers and their takes are one random draw, so the counts of stale takes referenced, of viewers who hold them, and of the anonymity sets would differ in another draw, though the structure would not. The anonymity sets are computed at the class level and treat observation of a slot as observation of the class, which a real observer may not have. The slider semantics, finally, are a proposal about what to show and none of the numbers tests whether viewers find them intelligible.

The next chapter takes the budget of a single viewer. Chapter 13 asks how a finite viewing life limits the number of editions and revisits that the ledger above will ever record for one person.

The scripts `lawnmodel.py` and `verify_ch12.py` reproduce each figure of this chapter.

Chapter 13

The Lifetime Repertoire

A viewer has a finite viewing life. The ledger of Chapter 12 will record, for one person, only as many editions as that life has room for, and the question this chapter asks is what the room is, how it is best spent among films, revisits, and release dates, and what a production must guarantee about the editions that fill it. The answer turns on a trade that the book has been circling since its first chapter. A person who wants never to see the same film twice can do so by seeing a great many different films, or by seeing a smaller number of films in editions that differ enough to count as different. The first costs productions. The second costs the tolerance of Chapter 2 and the capacity of Chapter 9.

The chapter works with the lawn scene and with the measured length of the passage of Chapter 6, an 87-minute film of 5,222 seconds. The viewing rates are assumptions chosen to be easy to change, and a person can have any rate. The space of possible films is never enumerated, and the chapter shows that it does not need to be.

13.1 The Budget

Let a viewer watch w hours per week for y years and let a viewing last h hours. The number of viewings is $N = \lfloor 52wy/h \rfloor$. With $y = 60$ and $h = 1.4506$, which is the length of the measured film, Table 13.1 gives the budget at three rates.

Hours per week	Lifetime hours	Viewings N	Viewings per year
2	6,240	4,301	71.7
6	18,720	12,905	215.1
14	43,680	30,112	501.9

Table 13.1: Lifetime viewings of an 87-minute film over sixty years at three weekly rates.

All later counts use the first row, $N = 4{,}301$. The budget is a bound on viewings, and it says nothing yet about which films fill it. A viewer who never revisits and wants no repeat of a film needs 4,301 different films, which is about seventy-two new films a year for sixty

years.

13.2 Revisits and the Novelty Horizon

A revisit is a viewing of a film already seen. Whether it is a repeat depends on the edition. If the edition shown is not perceptually equivalent to any edition the viewer has seen of that film, then by the standard of Chapter 2 the viewer has seen something new, though the story is the same. The *novelty horizon* of a production at tolerance ε is the number of viewings a single viewer can have of it with every edition pairwise more than ε apart. By the definition of the packing number in Chapter 2, it is M_ε. Two uses of viewing time both go by the name of revisiting, and the count should say which it is using. Chapter 1 budgets time for seeing an earlier film again, which is a repeat of the same work. This chapter counts a perceptibly different edition of the same work toward the novelty horizon, which is a new viewing of the work and not a repeat of what was seen. Both are legitimate uses of the budget.

Proposition 13.1 (Lifetime catalogue)**.** *Suppose every production in a catalogue has novelty horizon at least M at the viewer's tolerance. A viewer with N viewings can avoid any perceptual repeat within a production with $\lceil N/M \rceil$ productions, and cannot do so with fewer. At a constant viewing rate of N/y per year, the catalogue must grow at no fewer than $N/(yM)$ new productions a year to keep the viewer supplied.*

Proof. Each production supplies at most M viewings without a repeat, so $\lceil N/M \rceil$ productions are necessary, and they suffice because each can supply M. □

Table 13.2 gives the horizons of production A and production B after the agreement rule, from the packing numbers of Chapter 2, and the catalogue that a person watching two hours a week would need. The first row of each production is exact inequivalence, and the others are tolerances of one and two slots.

Prod.	ε	Horizon M_ε	Weeks to exhaust	Productions for 4,301 viewings	New films a year
A	0	238	172.6	19	0.30
A	1	54	39.2	80	1.33
A	2	16	11.6	269	4.48
B	0	96	69.6	45	0.75
B	1	32	23.2	135	2.24
B	2	12	8.7	359	5.97

Table 13.2: Novelty horizon and the catalogue a lifetime of 4,301 viewings needs, at 1.38 viewings a week. Without revisits the same viewer needs 4,301 films.

The compression is the ratio of the last column of viewings to the number of productions, and it equals the horizon. A viewer who accepts a tolerance of one slot in production A can fill sixty years with eighty productions in place of 4,301 films, and the production has to release one new film every nine months in place of one every five days. At tolerance two the horizon of A is sixteen, and a viewer who watches weekly exhausts a production's perceptually distinct editions in a little under twelve weeks.

This is the arithmetic that the framing of Chapter 1 asked for, and it comes with two qualifications that the table does not carry. A revisit is not the first viewing of a new story, and the proposition counts only the perceptual novelty of the edition. A viewer who remembers the film is not seeing the film for the first time at any tolerance, and Chapter 14 takes this up as the dependence of meaning on order. And the horizons are those of the toy scene, with a Hamming dissimilarity standing in for a measured one.

13.3 Selecting Without Enumerating

A repertoire is drawn from the editions of many productions, and for a real feature the class tuples cannot be listed. A feature with a thousand slots and twenty classes at each has a space that no enumeration could touch. Counting and selection do not need one.

The admissible class tuples under the direction rule form the paths through a layered graph, and the number of paths from each class at each slot to the end of the scene is computed in one backward pass: the number from a class at slot j is the sum of the numbers from the classes at slot $j+1$ that its direction accepts.

Proposition 13.2 (Unranking)**.** *Let the admissibility of a class tuple depend only on adjacent pairs, as the direction rule does. The number of admissible tuples is computed with $O(mq^2)$ additions of integers, for q classes at each of m slots, and the rth admissible tuple in lexicographic order is produced from r with $O(mq)$ comparisons. Every admissible tuple is produced for exactly one r.*

Proof. The table of counts decomposes the set of tuples by the class chosen at each slot in turn. At slot j, subtracting the counts of the classes passed over from r identifies the class whose block of tuples contains the rth, and the remainder indexes within the block. The map is a bijection between the integers below the total and the admissible tuples, since the blocks partition the set. □

The procedure was checked against full enumeration for all six pools of the lawn scene, the three productions at the premiere and after the agreement rule, and for every r it returns the same tuple as the enumeration: counts of 494, 238, 320, 96, 2 and 1. It was then run without enumeration on a synthetic scene of forty slots and ten classes, with a count of thirty-four digits, and on one of a thousand slots and twenty classes, whose count has 1,147 digits. Twenty ranks drawn at random in the larger scene were unranked into twenty distinct valid tuples in 0.16 seconds. The rules of Chapter 5 that reach past neighbors enlarge the table by the width of the frontier, which was one, two and three in the lawn scene, and the same argument applies.

A repertoire can therefore be a list of ranks. Selecting an edition is choosing an integer below the count, which a viewer's receipt can hold in the compact encoding of Chapter 12.

13.4 Scheduling a Repertoire

Selection has to meet two demands at once. At each moment the editions in the hands of the audience must be distinct, which is the demand of the earlier chapters. And over time each viewer's own sequence must avoid repeats for as long as the pool allows. A cyclic schedule over the ranks meets both.

Proposition 13.3 (Cyclic schedule)**.** *Let the admissible class tuples of a production be ranked* $0, \dots, K-1$*, let* $U \le K$ *viewers have distinct offsets* a_v *in the integers modulo* K*, and let* s *be coprime to* K*. If viewer* v *is shown rank* $(a_v + ks) \bmod K$ *at the* k*th round, then at every round the* U *viewers hold pairwise distinct editions, and every viewer sees each of the* K *editions exactly once in* K *rounds.*

Proof. At a fixed round the map from offset to rank is a translation of the integers modulo K, which is injective. For a fixed viewer the map from round to rank is a multiplication by a unit followed by a translation, which is a bijection on the K rounds. □

The proposition guarantees exact inequivalence only, and the perceptual quality depends on the stride. In ranked order, consecutive tuples are neighbors in the last slot. With stride one, a viewer's consecutive editions differ in 1.45 slots on average in A and 1.91 in B, and sometimes in one. A search over the strides coprime to K finds, for the 238 editions of A, stride 75 with consecutive editions differing in at least three slots and 4.18 on average, and for the 96 of B stride 47 with every consecutive pair differing in all five slots. These are distances between successive viewings and not guarantees that every pair of a viewer's editions is far apart, which would be the packing of the previous section and a shorter horizon.

The schedule also fixes what a tolerance costs for a simultaneous audience. The audience of sixty in A fits in the 238 editions of the exact promise, and does not fit in the packing number 54 at tolerance one. The audience of thirty in B fits at tolerance one, whose packing number is 32, and not at tolerance two, whose number is 12. A premiere to which everyone arrives at once can therefore promise separation by one slot in B and cannot in A, and the promise of Chapter 11 for locales has to be checked against the largest simultaneous audience in the same way.

13.5 What a Repertoire Must Guarantee

The preceding sections give three guarantees that a production can state about the editions that fill a person's life. The first is the *novelty horizon*, the number of viewings of one production before a perceptual repeat, which is the packing number of the corrected pool

and which Proposition 13.1 converts into a catalogue. The second is *simultaneous distinctness*, that no two viewers of a round hold equivalent editions, which is the Hall condition of Chapter 1 at the tolerance chosen and which Proposition 13.3 meets for the exact promise. The third is *survival*, that these guarantees hold through the corrections of Chapters 7 to 10. The schedule is drawn from the pool that survives every anticipated amendment, the invariant pool of Chapter 9, or else repaired as in Chapter 10 when an amendment arrives, and the repair rewrites the ranks of the viewers it moves. The no-repeat guarantee of the cyclic schedule does not by itself survive a reallocation: a replacement that is valid today can duplicate an edition the viewer watched yesterday. For the lifetime guarantee to survive corrections, the allocation needs each viewer's playback history as a constraint, with the perceptual neighbors of past editions excluded when the promise uses a threshold.

The ledger of one viewer is the lifetime form of the record of Chapter 12. In the compact encoding it holds 17 bits for each of the 4,301 viewings of production A, which is 73,117 bits or about nine kilobytes, and 14 bits and about seven and a half kilobytes for B. At fourteen hours a week the same ledger has 30,112 records and sixty-four kilobytes in A. These are the sizes at the premiere pool and would be smaller in the corrected pool. Production C has 28 bits a record and about fifteen kilobytes for a life, for a production with no perceptual choice at all.

13.6 What the Chapter Establishes and What It Assumes

The propositions hold for any pool, any tolerance, and any viewing budget. The counts rest on assumptions stated beside them: that a viewing lasts as long as the measured passage, that the viewing rates are examples, that every production in the catalogue has the novelty horizon of A or of B, that the packing numbers of Chapter 2 on a Hamming dissimilarity stand in for measured ones, and that a revisit counts as new whenever the edition is perceptually new, which a viewer who remembers the story will not grant. The cyclic schedule treats the audience as one cohort viewing in rounds, and a real audience arrives and leaves.

The next chapter drops the assumption that the films of a repertoire are items. Chapter 14 asks what a later film does to an earlier one, so that releasing the same film in the tenth year of a viewer's life may not be the same as releasing it in the first.

The scripts `lawnmodel.py` and `verify_ch13.py` reproduce each figure of this chapter.

Chapter 14

Sequence and Reinterpretation

The budget of Chapter 13 treats films as items. A viewer has so many viewings, a catalogue supplies so many productions, and the arithmetic is the same in whatever order the films are seen. Films are not items. A later film can change what an earlier one means, and what a film means at the first viewing is a fact about the viewer's life at that moment and cannot be had again. The order of release is therefore a variable, and a book about individual editions has to say what it varies.

This is the most open chapter of the book and it is written as a program. It states a model in which order can be seen to matter and in which it can be seen not to, computes what the model says on the engine scene of Chapter 3, and marks where the model stops and an experiment would have to begin. It does not claim that releasing a film in the tenth year of a viewer's life does something that releasing it in the first would miss. It states what would have to be true for that to hold.

14.1 Films as Commitments

The engine scene of Chapter 3 has 22 histories and four readings: diligence, negligence, disclosure, and concealment. An edition, by the account of that chapter, commits to a set of histories, those that agree with everything it shows. Let $C_f \subseteq H$ be the set committed to by a film f, which in this chapter may be an edition of any production set in the same story world. A viewer's *knowledge* $K \subseteq H$ is the set of histories not yet excluded by what the viewer has seen. The *reading set* of a knowledge state is the set of readings of the histories in it, and its size is the *openness*. A viewer who has seen nothing has all 22 histories and all four readings.

A viewing updates knowledge. Two updates are considered. The *intersection* update takes $K \cap C_f$ and is defined when that set is not empty. The *revision* update takes $K \cap C_f$ when it is not empty and otherwise takes C_f, so that a film that contradicts everything the viewer believed replaces it and the latest film has priority. Neither is a theory of how people understand stories. They are the two simplest ways of letting a later film act on an earlier one, and they behave differently.

Proposition 14.1 (Order and final state)**.** *For a family of films whose committed sets have a nonempty common intersection, every order of viewing leaves the viewer with the same knowledge, under the intersection update and under the revision update alike, namely $H \cap \bigcap_f C_f$. For two films whose committed sets are disjoint, the revision update leaves the viewer with the set of whichever film came last, and so the final state depends on the order.*

Proof. If the common intersection is nonempty, every partial intersection is nonempty, so neither update ever takes its second branch, and intersection is commutative. If the two sets are disjoint and nonempty, the second film's set replaces the first in either order, the two final states are the two different sets, and they are different because disjoint nonempty sets are never equal. □

The proposition separates two things that a discussion of reinterpretation tends to run together. The final knowledge of a viewer depends on order only when films contradict one another. What can depend on order in the other cases is the path. The *path* here is the sequence of knowledge states through which the viewer passed, and a summary of it, the openness of each film at its first viewing, is what the tables below report. Repeated or redundant films can give identical paths in different orders, so the claim is that paths can differ, and not that they differ in every order. The path is where the experience of a film lies.

14.2 Retrospective Reinterpretation

Take the 32 faithful editions of the engine scene, those consistent with the script history, with the distribution of openness in Table 14.1: eight leave one reading, ten leave two, eight leave three, and six leave all four, with a mean of 2.375.

Readings left open	1	2	3	4
Faithful editions	8	10	8	6

Table 14.1: Openness of the 32 faithful editions of the engine scene.

Now release a later film in the same world, and let the viewer apply it to what an earlier edition left open. Table 14.2 gives four later films by what they commit to. The first shows that a character noticed the bolt, and commits to 14 of the 22 histories. The second shows that a character noticed and concealed, committing to seven. The third shows the third bolt loose, committing to twelve. The fourth is unrelated and commits to all. A fifth, in the last row, contradicts the script by showing the third bolt tight, committing to ten.

A later film that commits to the concealment settles twenty-four of the thirty-two earlier editions. A later film that does not bear on the question leaves every reading set where it was. No later film in the table widens any reading set, and this is general.

Later film	Reading set shrinks	Mean openness, before → after	Inconsistent with the edition
Noticed	16 of 32	2.375 → 1.500	0
Noticed and concealed	24 of 32	2.375 → 1.000	0
Third bolt loose	12 of 32	2.375 → 2.000	0
Unrelated	0 of 32	2.375 → 2.375	0
Third bolt tight	0 of 24	2.500 → 2.500	8

Table 14.2: Effect of one later film on the reading sets of the faithful editions, by the intersection update. In the last row the eight editions that commit to the third bolt loose contradict the later film, and the means are over the other twenty-four.

Proposition 14.2 (Monotonicity)**.** *Under the intersection update, if $K' \subseteq K$ and both give a nonempty intersection with C_f, then the reading set of $K' \cap C_f$ is contained in that of $K \cap C_f$. Knowledge that grows can only narrow what an edition leaves open.*

The proof is that a subset of a set has the readings of a subset. A check over all 680 consistent pairs of a faithful edition with two of the later films found no counterexample. The consequence is worth stating for the question the chapter was asked. Within this model, a viewer who has seen more before an edition can find no reading in it that a viewer who has seen less could not. Everything that a later release gains over an earlier one has to come from somewhere the model does not yet contain.

14.3 Order and the Path

Take one of the six editions that leaves all four readings open, together with two later films, the noticing and the noticing and concealing. Three films can be seen in six orders. Table 14.3 gives, for each order, the openness at each film’s first viewing, taking each film’s reading set against what the viewer knew when it was seen.

The final knowledge is the same in every row, and the openness at first viewing is not. The last two orders give the same openness sequence, so openness does not separate every order, and whether the states visited differ in those two is a separate question that the table does not answer. The edition seen first is seen with four readings open, and seen after the concealment film it is seen with one. The total over the three viewings ranges from three to seven. Over all 32 faithful editions the spread of this total across the six orders is one for ten of them, two for fourteen, three for two, and four for six, which are the fully open ones, and the order that most often gives the largest total, in 24 of the 32 with ties counted, is the edition first, then the noticing, then the concealing. Order does not change what the viewer eventually knows, and it changes how long the question stays open, which is the part of a story that a viewer cannot repeat.

Order	Openness at each first viewing	Total
edition, noticed, concealed	4, 2, 1	7
edition, concealed, noticed	4, 1, 1	6
noticed, edition, concealed	2, 2, 1	5
noticed, concealed, edition	2, 1, 1	4
concealed, edition, noticed	1, 1, 1	3
concealed, noticed, edition	1, 1, 1	3

Table 14.3: Openness at the first viewing of each film in each of the six orders, for one fully open edition and the two later films, under the intersection update.

14.4 Contradiction and the Latest Film

When a later film contradicts the earlier, order does change the final state, and Proposition 14.1 says exactly when. Of the 32 faithful editions, eight commit to the third bolt loose, and a film showing it tight contradicts them. For those eight and that film the revision update gives the film's set when the film is last and the edition's set when it is first, and the final state depends on the order for those eight and for none of the other twenty-four. For the films that are consistent with every faithful edition, the noticing, the concealing, and the third bolt loose, the final state depends on the order for none of the thirty-two.

This is a small result and it fixes what the unreliable narrator costs in the model. A later film that contradicts an earlier one does not narrow the earlier reading. It replaces it, and the replacement stands only while no film follows it. The revision update gives the latest film priority and nothing in the model says that viewers do so.

14.5 Repertoires as Structures

If order matters to the path, a repertoire is a sequence and not a set. The space of repertoires of n films is then larger by a factor of $n!$: 120 orders of five films, 3,628,800 of ten, and a number with 119 digits for the eighty productions of Chapter 13. Release dependencies cut the space. A repertoire can be a partially ordered set, where one film is to be seen before another because the second presupposes the first, and the viewings that respect it are its linear extensions.

Each precedence halves or divides the orders, and a full chain removes the choice. A series built as one long chain offers a viewer one order, and a series built as independent episodes offers all of them. This is the structural trade of a repertoire: the more the films depend on each other, the more a later film can change an earlier one, and the fewer orders a viewer can choose among.

Five films, with the constraints	Admissible orders
none	120
the fourth film after the third (a reveal after its setup)	60
a chain of three, the others free	20
a chain of three and a chain of two	10
a chain of five	1

Table 14.4: Number of orders of five films that respect a set of precedence constraints.

14.6 The Tenth Year and the First

The question the outline poses is whether releasing a film to a viewer in the tenth year of life does something that releasing it in the first would miss. In the model of this chapter the viewer's knowledge at a given age is the intersection of the films seen before, so it only narrows with age, and Proposition 14.2 says the readings of an edition only narrow with it. A release in the first year sees at least as many readings open as one in the tenth. The model therefore answers that a later release can only lose openness, and what it gains must come from outside it.

Three possible gains are left as questions. A reading may require the viewer to know a premise, so that a viewer who lacks it cannot recognize the reading at all even though the histories that support it are not excluded. If so, the set of readings a viewer can recognize is not monotone in knowledge, and a later release can recognize what an earlier one could not. The premise of a reading, and whether viewers with and without it recognize the reading differently, is a measurement and the model contains no value for it. A second gain is that the viewer of the tenth year has different tolerances, those of Chapter 2, so that the same edition and a different viewer give a different equivalence. A third is that the first viewing also sets up the later, since a viewer who saw the film early has a memory of it that the later viewer lacks, and the memory is itself a film that the later release is read against. The model has no memory, because its viewer holds only a set of histories.

An experiment that would settle the first question would hold the edition fixed, vary the viewer's prior films, and ask which readings viewers report, for readings that have a premise and for readings that do not. If the second set is monotone in the prior knowledge and the first is not, the tenth year does something that the first misses. If both are monotone, it does not, and the release date is a matter of what the viewer wants to be told and when. The chapter does not make the prediction.

14.7 What the Chapter Establishes and What It Assumes

The propositions are theorems of the model: that order matters to the final state only through contradiction, that reading sets narrow with knowledge, and that the openness at first viewing can differ with the order. The counts are on the engine scene, a story of twenty-two

histories with four readings, and on five later films chosen by what they commit to. The two updates are the simplest available and neither is a model of understanding. A viewer here holds a set of histories and has no memory, no attention, and no forgetting, and a film commits to a set of histories exactly when its shots agree with them, which Chapter 3 defined and did not measure. The statement that a viewer's experience is a path of openness values is itself a modeling choice, since a viewer may not experience openness as the size of a set of readings.

The remaining chapters leave the single viewer and the toy scene. Chapter 15 asks what a production must exclude from its editions, and returns to counting.

The script `verify_ch14.py` reproduces each figure of this chapter.

Part V

Boundaries and Cases

Chapter 15

Exclusion and Reference Corpora

A production may declare material that its editions must not reproduce. The declaration might protect a competitor’s film, an earlier film of the same studio, or a performance that the studio has no right to repeat. The earlier chapters counted editions without regard to any such list. This chapter subtracts a list and asks what is left. It keeps the count apart from any determination of legal status. Whether an item is protected, whether two works are alike in the sense a court would apply, and whether a given edition infringes are questions that this book does not address and that no count in it answers.

The counts are taken on the lawn scene after the agreement rule, with the 238 class tuples of production A and the 96 of production B. An excluded item is a class tuple, and a reference corpus is a set of them. The corpora are drawn at random, either from the pool itself, which is the worst case for the subtraction, or from the whole product of classes, which is the case in which an outside corpus shares the vocabulary of the production but not its rules. Each count averages sixty draws.

15.1 A Policy and Its Three Parts

An exclusion policy has a corpus X, a representation ϕ in which an edition and an excluded item are compared, and a threshold ρ. An edition is *excluded* if some item of the corpus lies within ρ of it in the representation. The surviving editions are those outside every neighborhood.

The three parts are separate decisions, and the count depends on all of them. The corpus is a fact about the world. The representation is a choice about what counts as the same material, and the threshold is a choice about how close counts as too close. Exact subtraction is the case $\rho = 0$ in the finest representation, in which an item is excluded only if it is identical.

15.2 Exact Subtraction

If the representation is the class tuple, or the take tuple, and the threshold is zero, each excluded item removes exactly itself. The count falls by the number of excluded items that were in the pool, and nothing else changes. Excluding a thousand take tuples of production A by digest leaves 56,834 of 57,834, and the same thousand leave 2,072 of the 3,072 of production B.

The class tuple is a coarser unit than the take tuple, and one class tuple stands for several take tuples. In A every class has three takes, so excluding one class tuple removes $3^5 = 243$ take tuples. In B it removes $2^5 = 32$. Excluding ten class tuples removes 2,430 take tuples of A and 320 of B. The difference between the two units is the number of takes per class raised to the number of slots, and which unit a policy names decides how much it takes.

15.3 Approximate Subtraction

An approximate policy removes a neighborhood around each item. In the Hamming representation on class tuples, the neighborhood of radius ρ around an excluded tuple has, in the pool of A, a mean of 10.6 tuples at radius one, with a range of eight to fourteen, and a mean of 47.4 at radius two, with a range of thirty-two to seventy. In the pool of B it has exactly seven at radius one and exactly twenty-one at radius two. Radius two in a scene of five slots covers a fifth of the pool of A and a fifth of the pool of B around each excluded item.

Table 15.1 gives the survivors when the corpus is drawn from the pool, for n excluded items, under exact subtraction, under radius one and radius two, and under a coarser representation described below.

Production	Excluded items n	Exact	$\rho = 1$	$\rho = 2$	Direction pattern
A (238)	1	237	227.2	188.9	228.9
	5	233	189.5	78.0	198.9
	10	228	149.2	29.4	160.8
	20	218	92.9	3.3	119.7
B (96)	1	95	89.0	75.0	48.0
	5	91	65.1	26.4	2.4
	10	86	43.8	7.0	0.0
	20	76	16.8	0.3	0.0

Table 15.1: Mean number of surviving class tuples after n excluded items drawn from the pool, over sixty draws, by representation and threshold.

Twenty excluded items leave production A with about three of its 238 tuples at radius two, and leave production B with about seventeen of its 96 at radius one. A modest corpus under a modest threshold exhausts a small production. The corpus drawn from the whole

product of classes is less severe, because most of its items lie outside the pool: for $n = 20$ the survivors of A are 235.7, 196.3 and 48.8 at the three thresholds, and those of B are 95.6, 88.9 and 40.1.

15.4 The Representation Decides

The last column of Table 15.1 compares the editions by their direction pattern alone, which is the sequence of directions in which the character moves across the slots. Two editions are alike in this representation if they move the same way at every slot, whatever the footage. The pool of A has forty direction patterns, the largest shared by sixteen tuples. The pool of B has two, since B has no neutral class, and the direction rule forces every tuple to move all left or all right. Excluding one item of B in this representation excludes the forty-eight tuples that share its pattern, which is half the pool, and five items exclude all but 2.4 on average.

The representation is a statement of what is taken to be the same material, and a coarse one takes more. The example does not suggest that the direction pattern is a good representation of anything a policy would want to protect. It shows that with five excluded items the survivors of B differ by a factor of about forty between the finest representation and one that sounds modest, and that the choice has to be made, and defended, before the count means anything.

15.5 Exclusion and the Packing Number

The packing numbers of Chapter 2 bound how many editions can be pairwise distinguishable. Removing a neighborhood from the pool lowers them, and the amount is bounded when the neighborhood is small relative to the tolerance.

Proposition 15.1 (Packing under exclusion)**.** *Let $\mathcal{P}$ be a pool, let $\varepsilon \geq 0$, and let $B(x, \rho)$ be a Hamming ball with $2\rho \leq \varepsilon$. Then every set of editions that are pairwise more than ε apart contains at most one edition of the ball, and hence the packing number of $\mathcal{P} \setminus B(x, \rho)$ at tolerance ε is at least the packing number of $\mathcal{P}$ less one. Removing n such balls lowers it by at most n.*

Proof. Two editions of the ball are within $2\rho \leq \varepsilon$ of each other by the triangle inequality, so a set of editions pairwise more than ε apart contains at most one. A packing of $\mathcal{P}$ with its member in the ball removed is a packing of the remainder. □

The proof gives a more general bound. If the removed region can be covered by c balls of radius at most $\varepsilon/2$, then at most c points of any ε-packing lie in that region, and deleting them from a maximum packing leaves a packing of the survivors of size at least $M_\varepsilon - c$. The region may be a ball of any radius, since a larger ball can itself be covered by smaller ones. The difficulty is in getting a useful value of c from the representation and the exclusion policy without enumerating the pool.

The proposition is a bound, and the lawn scene shows it is not tight. The packing number of A at tolerance two is sixteen, and after removing balls of radius one around one, three,

five and ten excluded items it is still sixteen, against lower bounds of fifteen, thirteen, eleven and six. At tolerance one with exact subtraction it stays at fifty-four after ten exclusions. In B the packing number at tolerance two is twelve before exclusion, and twelve, eleven, ten and eight after one, three, five and ten, which is within the bound at each step, and at tolerance one with exact subtraction it stays at thirty-two. All of these were solved to optimality.

The reading is that a policy whose threshold is no more than half the tolerance of the promise costs the promise at most one distinguishable edition per excluded item, and in these examples much less. A policy with a larger threshold has no such bound, and the survivor counts of Table 15.1 show how quickly it can take everything.

15.6 Grammar Against Combination

Cinematic convention is reusable. A close shot of hands on a bolt, a wide view of a lawn, and a cut at a glance are not the property of any film, and a policy must be able to exclude a specific combination of imagery, performance, and timing without excluding the conventions from which every film is built. In the lawn scene a convention is a class at a slot, and a combination is a tuple.

Exact subtraction cannot damage a convention, since it removes tuples one at a time and the fewest tuples through any class in the pool of A is twenty-two, and sixteen in the pool of B. Approximate subtraction can, because a neighborhood removes every tuple through a class that its items cover. Table 15.2 gives, for the pool of A and of B, the mean number of classes left with no surviving tuple and the number left with under five percent of their tuples. The pool of A has nineteen classes over the five slots and the pool of B twenty-two.

Production	Threshold and n	Classes with none left	Classes below 5%
A	$\rho = 1$, $n = 20$	0.00	0.00
	$\rho = 2$, $n = 10$	1.53	5.80
	$\rho = 2$, $n = 20$	10.48	17.57
B	$\rho = 1$, $n = 20$	1.28	2.70
	$\rho = 2$, $n = 10$	10.40	13.47
	$\rho = 2$, $n = 20$	20.67	21.30

Table 15.2: Mean number of the classes of the pool that lose every tuple, and that fall below five percent of their tuples, after n excluded items drawn from the pool, over sixty draws. Exact subtraction loses none in either production.

At radius two, twenty excluded items leave production A with eight or nine of its nineteen classes still in use and leave production B with one of twenty-two. The policy that was meant to remove specific combinations has removed the grammar. A policy that must retain the conventions can state the requirement directly, as a minimum number of tuples that every class must keep, and check it after subtraction. The test is cheap, since it is a count

through each class of the survivors, and a policy that fails it has a threshold that is too wide for the scene.

15.7 The Count Apart From the Law

Every number in this chapter is a count of tuples under a declared policy. None of them is a determination that an excluded item is protected, that a neighborhood is a fair measure of likeness, or that an edition outside every neighborhood is free of any claim. The corpus is declared by someone, the representation is chosen by someone, and the threshold is set by someone, and the law's own standard of likeness does not have to coincide with any of the three. What the arithmetic supplies is the cost of a declaration: how much of a production a given policy takes, how few distinguishable editions it leaves, and whether it keeps the conventions from which films are made. That cost is the thing a production can know before deciding.

15.8 What the Chapter Establishes and What It Assumes

The proposition on packing holds for any pool and any Hamming ball. The counts rest on assumptions stated beside them: that the excluded items are class tuples of the same scene, that the Hamming dissimilarity on classes stands in for a measured one and a direction pattern for a coarse one, that the corpora are random draws and so show typical and not worst cases, and that a convention is a class at a slot, which is a convention of this toy and no more. A real corpus is not a set of tuples of the production's own vocabulary, and a real representation of imagery, performance, and timing is a feature extractor and a threshold that Chapter 2 posed and did not measure.

The next chapter turns from subtraction to generation. Chapter 16 asks what a generator must represent to produce takes under the same rules, and the rejection of samples in a generator is the exclusion of this chapter paid for in compute.

The script `verify_ch15.py` reproduces each figure of this chapter.

Chapter 16

Generation Under Constraint

The preceding chapters describe editions assembled from takes. The pool of a production has been a fixed list of classes, each with its takes, and an edition has been a path through it. This chapter asks what a generator must represent to produce takes on demand under the same rules, so that the pool is not fixed in advance. A generator has a support, which is everything it can produce, and rules, which say what it must not. The editions of a generative production are the elements of the support that the rules admit, and the count of the book is the size of that set.

The chapter treats three things. How a generator can be made to respect the rules, and what each way costs. How a correction reaches a generator that has already been built. And how a director's instructions, given at the level of a scene, a group of shots, and a single shot, narrow the set of editions in nested stages. The counts are on the lawn scene after the agreement rule, with the class tuples of earlier chapters standing for what a generator would emit. A generator of continuous footage has no such finite list, and the chapter states where that matters.

16.1 Support, Rules, and Correction

Let a generator emit an edition by choosing a class at each slot, and let its *support* be the product of the classes it can choose from, which in production A is $5 \cdot 5 \cdot 4 \cdot 4 \cdot 5 = 2{,}000$ tuples. The rules of Chapters 1 and 7 to 10 are constraints on tuples: the direction rule on adjacent slots, and the agreement rule removing stale classes. The *admissible set* is the part of the support that satisfies the rules, and for A it has 494 tuples at the premiere and 238 after the agreement rule. A correction, in this setting, changes the rules, and the generator has to be told.

There are three ways to make a generator respect the rules. It can propose from the whole support and reject what violates them. It can build its proposal slot by slot, allowing at each slot only what agrees with the slot before. Or it can carry a table that counts the admissible completions of every partial edition and sample from the table, so that it never proposes a violation. The next three sections cost them in turn.

16.2 Rejection

If proposals are uniform over the support, each is admissible with probability equal to the ratio of the admissible set to the support, and the number of proposals before an acceptance has expectation the reciprocal of that ratio. Table 16.1 gives the ratio for each production at the premiere and after the agreement rule.

Prod.	Support	Admissible, premiere	Admissible, after rule	Proposals per accepted edition, premiere $\to$ after rule
A	2,000	494	238	4.0 $\to$ 8.4
B	5,880	320	96	18.4 $\to$ 61.2
C	32	2	1	16.0 $\to$ 32.0

Table 16.1: Cost of rejection sampling from the class product. The expected number of proposals is the support divided by the admissible set.

A correction that removes admissible tuples raises the cost of every acceptance by the same ratio, so the cost of rejection is the cost of the correction paid on every future edition. In B it is sixty-one proposals for each edition delivered. The cost has a second part when an audience is served. To give sixty viewers of A sixty different editions by drawing from the corrected set takes 69.0 draws in expectation, since later draws collide with earlier ones, and rejection from the support takes about 580 proposals for it. For the thirty viewers of B the draws are 35.7 and the proposals are 2,189.

16.3 Local Constraints Without Lookahead

A generator that builds an edition from left to right and checks each slot against the one before it does not reject whole editions. It chooses uniformly among the classes that agree with the preceding choice. The probability that it produces a given tuple is the product, over the slots, of the reciprocal of the number of classes that were allowed at that slot, and that product depends on the tuple.

Proposition 16.1 (Local sampling)**.** *A left-to-right generator with no lookahead produces each admissible tuple t with probability $\prod_j 1/n_j(t)$, where $n_j(t)$ is the number of classes allowed at slot j after the choices that t makes before it. It is uniform over the admissible tuples that it produces if and only if this product is the same for all of them. It fails, with the remaining probability, whenever some prefix leaves no class allowed at the next slot.*

The proposition is immediate from the definition and the lawn scene shows what it does. Table 16.2 gives, for each production at the premiere and after the rule, the probability that the generator reaches a dead end, the total variation distance of its output from the

uniform distribution over the admissible set, and the factor by which it over-samples the most favored tuple.

Production	Stage	Dead end	Distance from uniform	Most favored tuple
A	premiere	0.000	0.243	2.74 times
A	after rule	0.000	0.186	2.64 times
B	premiere	0.000	0.150	1.60 times
B	after rule	0.000	0.100	1.20 times
C	premiere	0.500	0.000	1.00 times
C	after rule	0.500	0.000	1.00 times

Table 16.2: Left-to-right sampling without lookahead, computed exactly by enumerating its paths. Distances are over the tuples that the generator completes.

In A and B the generator never fails and yet it is not fair: after the agreement rule its output is 0.19 away in total variation from the uniform distribution on A’s 238 tuples, and it produces one tuple 2.64 times as often as uniform would. A bias of this kind matters to a cinema of individual editions, because the count of the book assumes that editions are handed out without favoring any, and a favored tuple is an edition given to more viewers than its share. In C the generator fails half of the time, because its first choice can leave no class allowed at a later slot. A generator without lookahead is thus either biased or incomplete in every production here, and not both in the same one.

16.4 Exact Sampling by Table

The table of Chapter 13 removes both defects. It counts, for each class at each slot, the number of admissible completions, and a generator that chooses each class with probability proportional to its count produces each admissible tuple with equal probability and never reaches a dead end. This is uniform sampling by unranking, with a random rank drawn from the admissible count. It costs one table, built once, and $O(mq)$ operations per edition.

A correction changes the table, and the cost of the change is the cost of rebuilding it. The table needs one compatibility check for each pair of live classes at adjacent slots. In A the check count falls from 81 before the agreement rule to 53 after it, and in B from 128 to 76. These are the costs of telling an exact generator about a correction, and they are paid once per correction, in place of a factor on every edition as in rejection.

The saving has a limit. The table represents rules that act on adjacent slots, and the rules of Chapter 5 that reach further widen the table by the width of the frontier. A rule that cannot be stated as a constraint on a bounded neighborhood, such as a rule on the whole edition, has no table of this kind unless it admits a compact sufficient state. Some global predicates do: a parity, or a bounded counter of how many times a class has been used, can be carried along the slots as extra state, and the table is then indexed by that state as well.

The computational question is whether the rule has a manageable representation across the frontier. A rule with none brings the generator back to rejection.

16.5 Generating a Distinguishable Audience

Chapter 2 asked for editions more than a tolerance apart. A generator can meet it by proposing and rejecting, keeping each proposal that is far enough from those already chosen. This is random greedy packing, and it stops when no admissible tuple remains that is far enough. Table 16.3 gives the size of the set it reaches over 500 random orders, against the optimal packing number of Chapter 2.

Production	Tolerance	Optimum	Mean reached	Range	Optimal in
A	1	54	49.6	43 to 54	3 of 500
A	2	16	13.5	10 to 16	12 of 500
B	1	32	28.7	23 to 32	101 of 500
B	2	12	8.5	8 to 12	15 of 500

Table 16.3: Size of the separated set reached by random greedy selection from the corrected pool, over 500 random orders.

A generator that selects as it goes reaches about nine tenths of the optimum in A at tolerance one and about seven tenths in B at tolerance two, and hits the optimum rarely. The consequence for a production is that a tolerance promised from the packing number of Chapter 2 will not be delivered by a generator that selects at random. A generator that is to deliver it needs a designed set of editions, as in the codes of Chapter 11, and not a stream of draws.

16.6 Hierarchical Direction

A director gives instructions at several levels. A command about the scene constrains every slot. A command about a group of shots constrains some of them. A command about a shot fixes it. Nested commands narrow the admissible set in stages, and each stage is a restriction of the domains of slots, so the table of the previous sections counts the editions that remain.

Take production A after the agreement rule, and four nested commands. The scene command forbids rightward movement at any slot. The group command requires that the character move at the second slot. The next command fixes the first slot to its first remaining class, and the last fixes the third slot in the same way. Table 16.4 gives the number of editions after each command, for A and for B.

In A the scene command alone leaves 54 editions for sixty viewers, so the first instruction a director gives already makes the exact promise impossible. In B the audience of thirty fits until the first shot is fixed, which halves the pool to twenty-four. A command that is

	None	Scene	Group	Shot 1 fixed	Shot 3 fixed
A (audience 60)	238	54	36	12	12
B (audience 30)	96	48	48	24	8

Table 16.4: Editions remaining after each nested command, by the table of the previous sections.

harmless to the story can be fatal to the capacity, and a generator that accepts commands should check the count after each one, which the table does at no extra cost. A command that restricts the domain of a slot is an intersection, so two such commands give the same pool in either order. The last two commands of the example, which fix a slot to its first remaining class, depend on what the earlier commands left, and that is a convenience of the example.

16.7 What the Chapter Establishes and What It Assumes

The statements on rejection, local sampling, and the table hold for any rules that act on adjacent slots, and the proposition on local sampling holds for any pool. The counts rest on assumptions stated beside them: that a generator emits class tuples, so its support is a finite product, whereas a generator of continuous footage has a support that cannot be counted and for which the cost of rejection is not the ratio of two finite sets; that the proposals are uniform, whereas a trained generator proposes from a distribution that concentrates, so the acceptance rates here are those of a naive proposal and not of any real model; and that the commands of the table were chosen to show the narrowing, and not as those a director would give. How a trained generator is told of a correction, by retraining, by masking at proposal, or by constraining its decoding, is not computed here, because nothing in this book is a trained generator.

The next chapter carries the machinery through three complete studies. Chapter 17 follows the engine scene, the lawn and a dishwashing scene, and a measured passage of an existing film through a correction to their records.

The script `verify_ch16.py` reproduces each figure of this chapter.

Chapter 17

Case Studies

Three studies carry the machinery of the book through complete examples. The first follows the engine scene of Chapter 3 through three local productions and a delayed recognition. The second follows a correction through the full ledger of a film that has two scenes, the lawn and a dishwashing scene. The third takes a measured passage of an existing film and follows the acoustic occupation record of Chapter 6 through a correction to its own reference. Each study states what the theory predicts, what it does not determine, and what an experiment would have to settle.

The dishwashing scene appears in no earlier chapter, and its pools are invented here for the purpose, with the invention stated where it is used. The other two studies use the scenes and data of the earlier chapters unchanged.

17.1 Study One: The Engine Scene in Three Locales

The engine scene has 126 class tuples, of which 32 are faithful to the script history, and four readings that a viewer may reach: diligence, negligence, disclosure, and concealment (Chapter 3). Release it in three locales, with local trace in the close shots and the handoff, so that the editions of different locales are different global classes (Chapter 11), and give each locale thirty viewers, drawn at random from its faithful editions. The audience is ninety, and the three locales together have 96 faithful global editions, so the exact promise fits. An audience of forty in each locale would need 120 and does not.

Delayed recognition. The faithful editions leave different numbers of readings open, and so do the viewers who hold them. Table 17.1 gives the ninety viewers by the number of readings their edition leaves open at the first viewing. Twenty-one are settled from the start, and sixty-nine hold an edition that leaves at least two readings open. A later film that commits to the concealment, in the sense of Chapter 14, narrows the reading set of each of those sixty-nine, and leaves all ninety with the reading of concealment alone.

What two viewers can discuss. The opening argument of the book is that a shared premiere still gives viewers something to talk about, even when their editions differ. The engine scene can say how much. Take the 32 faithful editions as the editions of one locale,

Readings left open	1	2	3	4
Viewers (of 90)	21	29	24	16

Table 17.1: Ninety viewers of the engine scene, thirty in each of three locales, by the number of readings their edition leaves open at the first viewing.

so that two viewers who see the same class see the same shot, and ask what each viewer is warranted in saying about the third bolt, the claim that it was left loose. An edition warrants a claim when every history it commits to satisfies the claim. Before any reveal film, 8 of the 32 editions show the loose third bolt directly, none warrants the claim by inference from other shots, and 24 leave it open. Of the 496 pairs of viewers, 28 have both seen it, 192 have one who has seen it and one who has not, and 276 have neither. The evidence that two viewers can point to in conversation is thinner still: counting only the shots that bear on the story, the close-ups, the glance, and the handoff, 233 of the 496 pairs have no informative shot in common, 205 have one, 55 have two, and 3 have three.

The reveal film changes what is shared and leaves the rest. A film that commits to the concealment settles the reading for all 32 viewers, so every pair now shares the claim that someone skipped a bolt and concealed it, against 28 pairs before. It does not settle which bolt. Only the 8 viewers who were shown the third bolt loose still warrant the claim about it, and in 468 of the 496 pairs the viewers share the reading without both warranting the event. Three things are therefore different. Two viewers can share an event, in that both warrant the same claim about what happened. They can share evidence, in that both can point to the same shot. And they can share the experience of discovering it, which is the path of Chapter 14 and which the openness at first viewing in that chapter summarizes. The model counts the first two and does not measure the third. The script `verify_ch17_shared.py` reproduces these figures, and what viewers would actually say to one another remains an empirical question.

A wrongful correction. A viewer reports that the close shot of the third bolt shows it loose. By Chapter 7 the report is rejected when the departure is intended, as it is here, since the script history has the third bolt loose. Suppose it is instead upheld as an error and the class is removed. Each locale has 24 faithful editions left, against an audience of thirty, and the shortfall is six in each locale. The class had been held by eight viewers in the first locale and seven in each of the others, and the free editions left over to move them to are two, one, and one. Eighteen of the ninety viewers cannot be given a distinct faithful edition in any repair. The editions that recover the concealment alone fall from eight to six.

What the theory predicts. It predicts the shortfall exactly, since the capacity of each locale is a count and the audience is another, and it predicts that the failure falls on the viewers who held the removed class. It predicts that the reveal film settles the reading for every viewer, since each faithful edition is consistent with the script history. It predicts that a rightful rejection of the report costs nothing.

What it does not determine. Whether a viewer in the sixty-nine finds the film open in the

way the model counts, so that a settled viewer sees a different film. Whether three locales really differ at the close shots and the handoff, which was assumed. And what the viewers of a locale would do if told that eighteen of them share an edition.

What would settle it. A reading task with the faithful editions of one locale, with and without the reveal film beforehand, would show whether the reading sets of the model predict what viewers report. A comparison of the close shots across locales, by viewers who are asked to say whether the shots differ, would test the assumed local trace.

17.2 Study Two: A Correction Through a Ledger

Take a film of two scenes, the lawn of production A and a dishwashing scene with four slots and, in order, three, three, two and three classes, two takes to a class. The pools of the dishwashing scene are invented for this study. At its third slot one class shows a plate clean before it has been washed, which is the same kind of continuity error that the agreement rule removes from the lawn. The editions of the film are pairs of a lawn edition and a dishwashing edition.

At the premiere the lawn has 494 editions and the dishwashing scene 54, so the film has 26,676 and, with takes, 103,716,288. After both corrections the lawn has 238 and the dishwashing scene 27, and the film has 6,426 editions and 24,984,288 with takes. The audience is five hundred viewers, each with a distinct pair, drawn at random from the premiere product.

The two corrections reach the ledger in sequence. Table 17.2 gives what each does. The lawn correction invalidates 263 of the five hundred pairs, the dishwashing correction 228, and both together 370, which is the three quarters of the audience whose pair held a stale class in either scene.

Step	Pairs invalid	Radius	Cost	Records added
Lawn correction	263	2	310	263
Dishwashing correction	228	2	237	228
Total, in sequence	370 viewers moved		547	491
One shot, both together	370	3	539	370
Anticipated allocation	0		0	0

Table 17.2: A correction through a ledger of five hundred viewers of the two-scene film. Radius and cost are Hamming over the nine slots, survivors fixed, minimum total movement.

In sequence, 121 of the 370 viewers who move are moved twice, because a viewer placed by the first repair can land on a pair that the second removes. The ledger grows from 500 records to 991. A single repair of the two corrections together moves the same 370 viewers for a slightly lower cost, 539 against 547, at a larger radius, three against two, and adds 370 records, so the ledger has 870. Allocating at the premiere from the pool that survives both

corrections, which has 6,426 pairs for five hundred viewers, moves no viewer and leaves five hundred records.

The archive has to keep every stale take that the ledger names. The five hundred viewers name all twelve stale takes of the lawn and both stale takes of the dishwashing scene, and the 370 viewers whose editions held a stale class cannot be reconstructed if those takes are purged.

What the theory predicts. The counts of this section follow from the pools and the allocation, and the comparison of sequence and one shot follows from Chapter 10: the order in which corrections arrive changes the cost and the number of moves by viewers, and the final pool does not depend on it. The theory predicts that anticipating the corrections requires no later movement of any viewer here, because the surviving pool is nearly thirteen times the audience. It is not free: anticipation restricts the premiere catalogue from 26,676 editions to 6,426, which is the trade that the capacity of Chapter 9 describes.

What it does not determine. That the two corrections would arrive as separate events, which is a fact about a production's workflow. That a viewer who has been moved will be told, and what such a viewer's receipt should show, which Chapter 12 raised as a design choice and no count settles. That the dishwashing scene's continuity error has the structure assumed, since its pools are invented.

What would settle it. A production's own correction log, replayed against a ledger of its real audience, would give the numbers of the table for a real film. The comparison with anticipated allocation would then be a measure of what the production left on the table.

17.3 Study Three: A Correction to the Record of a Measured Passage

The acoustic occupation record of Chapter 6 is a measured account of one existing film, and the dialogue reference is one of its entries. It is the median of the level over the 1,499 seconds that the subtitles define as dialogue, which is -34.554 LUFS, and everything else in the record is read against it: a second is occupied when it is more than ten decibels above it, a block is an occupied run of at least sixty seconds that merges gaps of up to five, and a recovery window is a run of at least five seconds at or below it or inaudible. With the recorded reference the film has nine blocks, 1,004 seconds of them, and 105 windows, 1,526 seconds, and the passage of Chapter 6 has windows of 9, 7 and 13 seconds and six blocks.

Suppose the reference is corrected. Table 17.3 gives the recomputed record for a shift of the reference by amounts that correspond to real alternatives in the record: the proxy reference from 1,509 proxy seconds, which is 0.44 decibels lower, the reference taken from the centre channel, which is 2.33 decibels lower, and shifts of the opposite sign of the same size.

The record is sturdier in one direction than in the other. Lowering the reference by the proxy's 0.44 decibels leaves the passage exactly as it was, with the same three windows and the same slack of fourteen seconds, and lowering it to the centre-channel value takes

Reference shift	Blocks	Windows	Passage windows (s)	Slack (s)	Blocks in passage
−2.33 dB (centre)	9	85	5, 5	0	5
−1.00 dB	9	100	9, 5, 9	8	6
−0.44 dB (proxy)	9	100	9, 7, 13	14	6
0 (recorded)	9	105	9, 7, 13	14	6
+0.44 dB	7	98	9, 7, 13	14	5
+1.00 dB	6	91	10, 12, 13	20	4
+2.33 dB	2	94	6, 8, 10, 27, 6, 7, 9	38	2

Table 17.3: The occupation record of the measured film recomputed under shifts of the dialogue reference, with the rest of its definition unchanged.

the slack to zero, since the passage then has two windows of five seconds with nothing to shorten. Raising it by the same 0.44 decibels removes two of the nine blocks, one of them in the passage, and raising it by 2.33 leaves two blocks. A half decibel is a small difference in a measurement of loudness and an entry in the record that decides whether a passage has a recovery pause worth preserving.

The reference is also sensitive to the alignment of the subtitles on which it rests. Shifting the dialogue seconds by minus three, minus one, plus one, plus three and plus ten seconds moves the median by -0.45, -0.35, -0.65, -1.05 and -0.38 decibels, downward in each of the five, and an offset of plus three seconds already changes the passage's windows to 9, 5 and 9 with a slack of eight. A misaligned subtitle file therefore corrects the record in the direction that lowers the reference, and a lowered reference makes the film's blocks longer and its windows fewer.

What the theory predicts. The timing editions of Chapter 6 inherit the record. Slack bounds how far a cut can shorten a window before it stops qualifying, so a record whose passage slack goes from fourteen seconds to zero removes the room for any shortening, and a record whose slack goes to eight leaves a little. The corrected record changes the count of timing editions that stay within the window tolerance, though the counts of Chapter 6 for the slot handles do not change.

What it does not determine. Whether the recorded reference, the proxy, or the centre-channel value is the correct one, which depends on what a viewer takes to be the level of dialogue. Whether a window of five seconds at or below the reference is a pause that a viewer experiences as recovery, which the threshold of five seconds assumes and the record does not test. And whether two of the nine blocks that the upward shift removes were ever occupied in the sense of a listener.

What would settle it. A listening experiment on the passage, with the pauses of the three windows shortened by amounts within and beyond the slack, and with the blocks

listened to at the recorded and the shifted reference, would show which of the records predicts what listeners report. The experiment would choose among the references, and so correct the record by a measurement. It is a different thing from the recomputation of the table. Changing the analytical reference changes the annotation of a fixed recording, whereas editing the sound tests what a listener experiences, and only the second can say which annotation is right.

17.4 What the Studies Show and What They Assume

The first study shows that a count settles whether a correction can be repaired and who is left without an edition, and that a wrongful correction costs a whole locale its promise. The second shows that a correction to two scenes costs about three quarters of a film's audience a move and leaves a ledger of twice the length, and that a pool that survives every correction would have cost nobody anything. The third shows that the record on which a timing edition rests is itself sensitive to a half decibel and to a few seconds of subtitle offset. None of the three is an experiment, and in each the last paragraph says what the experiment would be.

The assumptions are those of the earlier chapters, restated for each study where it matters: the Hamming dissimilarity on class tuples, the local trace of the locales, the invented pools of the dishwashing scene, the median reference and the thresholds of the occupation record. The studies were chosen to cover a delayed reading, a ledger, and a measurement, and a different choice of scene would have shown different numbers.

The last chapter of the book collects what remains open.

The scripts `lawnmodel.py` and `verify_ch17.py`, with the data of Chapter 6, reproduce each figure of this chapter.

Chapter 18

Open Problems

This chapter collects what the book leaves undecided. It is ordered by what each problem would unlock, and it keeps two kinds of question apart. The mathematical problems are about the models of the preceding chapters and would be settled by proof or by computation. The empirical problems are about viewers, footage, and productions, and no amount of mathematics settles them, however many tables the earlier chapters contain. The chapter introduces no new counts.

It also has to say what the book does not show. It does not show that a cinema of individual editions is desirable, feasible, or something that anyone should build. Every capacity in it is a property of a toy scene of five slots, every tolerance is a stand-in for a threshold that nobody has measured, and the rate of Chapter 1 has a denominator that has never been computed. The arithmetic is exact, and it is exact about scenes that were chosen so that it could be. The book does not support its own premise, and the problems below are the reasons.

18.1 Mathematical Problems

Exact covering numbers. Chapter 2 gives the packing numbers of the corrected pools exactly and the covering numbers as ranges where the solver stopped: for production A at tolerance one the covering number lies between 25 and 31 and was not resolved. A covering is the smallest set of editions such that every edition is within the tolerance of one of them, and so it is the number of editions needed to represent the pool at that resolution. It is related to the packing number by the inequalities of Chapter 2 and is not otherwise its dual: a packing governs the simultaneous separation of viewers, and a covering governs representation within a tolerance. It is the quantity that a compression claim would use. Closing the gap is a finite computation on a pool of 238 tuples, and a structural argument that gave the covering number for a class of pools would be better than the computation.

Rules that reach past neighbors. Counting and unranking in Chapters 13 and 16 depend on the admissibility of a tuple being decided by adjacent pairs, or by a bounded frontier as in Chapter 5, where the frontier widths of the lawn scene are one, two, and three. A rule on the whole edition has no table and sends a generator back to rejection. The open question is which families of rules keep counting and unranking polynomial, and which make counting hard. A dichotomy of the kind known for constraint satisfaction problems is the form of answer one would want, and none is claimed.

Computing robust capacity. Chapter 9 defines robust capacity as the minimum over the amendments in a family of the capacity left after each, with a separate allocation for each, and invariant capacity as the capacity of the intersection. Both were computed by enumeration on the lawn scene. The complexity of computing them for a family of amendments given as rules, without enumerating the pool, is open, as is which structured families of amendments admit a useful bound between the two. No bound holds for arbitrary families: two survivor sets of size k with one class in common have robust capacity k and invariant capacity 1, and with no class in common invariant capacity 0. In the lawn scene they differ by a factor of about one and a half for production A, 351 against 238, and a general bound would say when anticipating a correction is as good as waiting for it.

Repair under repeated amendment. Chapter 10 treats repair as a minimum-cost assignment of invalid viewers to free valid editions, and the radius, the cost, and the number of moves are three different objectives. In production B the repair of minimum total movement has radius three, while a repair of radius two exists, and in sequence the cost is 28 against 27 for the one-shot repair. The open problem is the online one: when amendments arrive without notice and each must be repaired before the next is known, how much more than the offline optimum must be paid, in cost and in the number of times a viewer is moved. No bound is proved, and the lawn scene shows only that the gap can be nonzero.

Worldwide footage with local and shared slots. Chapter 11 proves that locale-by-locale allocation is sound for tolerances below the number of slots that carry a local trace, and gives lower bounds, from the Singleton bound, on the classes per slot that a locale needs. The bounds for an audience of 238 were not checked for attainment, and the problem of how to divide footage between local and shared slots so as to minimize total footage, subject to the Hall condition across locales and a separation tolerance, is posed and not solved. Codes over mixed alphabets with some coordinates shared between several codes are the likely tool.

Designed schedules. The cyclic schedule of Chapter 13 gives every audience distinct editions in every round and every viewer the whole pool without repeat, and it guarantees exact inequivalence only. Chapter 16 shows that a generator that selects at random reaches roughly seven tenths to nine tenths of the optimum packing in the cases tried and seldom reaches the optimum. What is wanted is a schedule that gives simultaneous separation by

a tolerance and, for each viewer, a sequence of mutually separated editions for the whole novelty horizon. The difficulty is that the pool is a product with tuples removed by rules, and the algebraic constructions that give optimal codes assume a full product. Constructions inside constrained pools are open.

The class of order-independent updates. Chapter 14 gives two updates, intersection and revision, proves that the final state is independent of order when the committed sets have a common intersection and dependent on order for two films whose sets are disjoint, and shows on the engine scene that the openness at first viewing can depend on order. Families between the two cases, such as films that overlap pairwise without a common intersection, are not characterized, and neither are the updates that are order-independent in the final state or in the whole path. It would say which models of understanding admit a book's worth of films read in any order, and which do not.

Exclusion beyond half the tolerance. The bound of Chapter 15 removes at most one packing point per ball when the neighborhood radius is at most half the tolerance, and the covering form of it gives $M_\varepsilon - c$ for a removed region covered by c balls of radius $\varepsilon/2$. The survivor counts show that a larger radius can take the whole pool quickly. What is open is how to obtain a useful value of c from the representation and the exclusion policy without enumerating the pool, as is the complexity of deciding whether a policy retains every class of the grammar above a stated fraction.

When local sampling is fair. Chapter 16 shows that a left-to-right generator without lookahead is uniform over its output if and only if the product of option counts is the same for all admissible tuples. Which families of rules have this property is not characterized. The rules of the lawn scene do not, and a result that said which rules do would say when a table is unnecessary.

Anonymity against capacity. Chapter 12 shows that a ledger of unique editions is a ledger of near-identifiers. A production could promise instead that each edition is shared by at least k viewers, which relaxes the capacity requirement from the audience to the audience over k and restores an anonymity set of k. The trade between the individuality that the book promises and the anonymity that its ledger needs is a one-parameter family that the book states and does not study. What the least pool is for a given k and tolerance, and how far the promise of individuality survives at $k > 1$, is open.

18.2 Empirical Problems

The storage cost. The rate of Chapter 1 is $\log_2 K/B$, with B the cost of the footage, rules, and metadata needed to reconstruct the editions. The book computes K and never B. The claim that individual editions are a better compression of film than a single edition is

therefore not tested by anything here, and it would be tested by the measurement of B for one real production, with the perceptual distinctions among its editions established, and by a baseline: the storage needed by two representations to deliver the same catalogue, or the same perceptual coverage at a stated tolerance, measured at the same revision. This is the first problem in the order of what it would unlock, because the rate is the justification of the framing and it has a missing denominator.

The thresholds of perceptual equivalence. Chapter 2 defines three pseudometrics, for detection, comprehension, and affect, and the tolerances at which two editions count as the same are unmeasured. Every packing number, every novelty horizon, and every count of distinguishable editions in the book depends on them. A measurement would give a threshold for each kind of difference and each kind of viewer, and would say whether the Hamming dissimilarity on class tuples that the book uses is anywhere near the measured one.

Local trace. Chapter 11 assumes that local footage at a slot differs perceptibly between locales and that shared slots do not. Whether it does is a fact about footage and viewers. A comparison of takes across locales, by viewers who are asked whether they differ, would settle which slots carry local trace and so how far locale-by-locale allocation is sound.

Shared viewing. The exchange of Chapter 1, in which two viewers both say that the same character missed a strip of lawn, one from a close shot and one from inferring it from the finished lawn, is the reason the book gives for wanting editions to be watched together. Whether viewers who have seen different editions can have such an exchange, how much of the film they can discuss, and what they take to be the film, is not known. It is the question on which the practical value of the whole programme depends.

Openness as experienced. Chapters 3, 14 and 17 count the readings that an edition leaves open and treat the order of films as changing how long a question stays open. Whether viewers experience openness as the size of a set of readings, whether a revealed reading is recognized as the one that was left open, and whether a viewer who is settled at the first viewing sees a different film, are questions for a reading task of the kind Study One of Chapter 17 describes.

Recognition and the tenth year. Chapter 14 shows that in its model a later release can only lose openness, and leaves open whether a reading that requires a premise is recognized differently by viewers with and without it. The experiment that would settle it holds the edition fixed, varies the viewer's prior films, and asks which readings are reported. Without it the question whether a release in the tenth year of a viewer's life does something that a release in the first would miss has no answer.

Recovery windows and the dialogue reference. Chapter 6 and Study Three of Chapter 17 take the occupation record at its threshold of five seconds and its reference of −34.554 LUFS, and show that a shift of 0.44 decibels in either direction leaves the passage's three windows unchanged, although the upward shift reduces the film's nine blocks to seven, and that a downward shift to the centre-channel value of 2.33 decibels leaves two windows of five seconds with zero slack, which differs from having no windows. Whether a window of five seconds at or below the reference is experienced as recovery, which of three references is the level of dialogue, and whether the saturating response of Chapter 6, which is a stipulated model, describes viewers at all, are listening questions.

How corrections arrive. The ledgers of Chapters 10, 12 and 17 assume corrections that arrive as discrete events with a known rule, and the claim rates of Chapter 7 are assumed. How real reports arrive, how many are true, how long adjudication takes, and whether the amendments of a real production have the shape of the three of the lawn scene, would come from a production's own log replayed against a ledger of its audience.

What viewers disclose. The anonymity sets of Chapter 12 are computed at the class level, from an observer who knows the class of a viewer's edition at some slots. How much of an edition viewers reveal in talking about a film, and to whom, is unknown, and so is the number of slots an observer typically has. The count says what an observation of s slots leaves, and not how often one occurs.

Whether it makes viewing together easier. The larger aim of the programme is that individual editions might one day lessen the need to watch films all together, and make a society more interesting and easier to move around in. The book contains no evidence about any society. Whether individual editions make shared experience richer, or poorer, or neither, depends on how people who have seen different films talk to each other, and that is learned from people and not from capacity.

18.3 What Each Would Unlock

The problems are not equally urgent. The measurement of B and the measurement of the thresholds of equivalence come first, because the rate and every count of distinguishable editions are conditional on them, and until they are made the book's central quantities are the numbers of a toy. The question of shared viewing comes next, since it decides whether the programme has a purpose that survives the toy. The design problems, for schedules inside constrained pools, for footage across locales, and for anticipated allocation, come after, because they are only worth solving at the thresholds that the measurements give. The remaining mathematical problems, on covering numbers, on rules beyond neighbors, on online repair, and on the class of order-independent updates, are independent of the measurements and can be pursued at once. The experiments on openness, recognition, and

recovery windows are the ones that connect the model of meaning to viewers, and they unlock the chapters on sequence and sound.

None of them is begun here. The book states what it counts, what the counts assume, and what would have to be found out, and it does not claim to have found any of it out.

18.4 What the Chapter Establishes and What It Assumes

The chapter establishes only a list, and the list is incomplete: it contains the problems that the author of the book met while writing it, and a reader will meet others. The order is a judgment about unlocking, and a different judgment would give a different order. The separation of the mathematical from the empirical is cleaner on the page than it is in practice, since several of the mathematical problems, such as the choice of footage across locales, are posed with parameters that only the empirical problems can supply.

No script accompanies this chapter, since it contains no count.

Appendix A

Counting and Matching

This appendix gives the finite mathematics that the text uses, with complete proofs for the cases the text needs. It covers transfer matrices, ranking and unranking, Hall's theorem and its worldwide form, minimum-cost assignment, and packing and covering numbers. Where a result is cited and not proved, the appendix says so. Each statement that the text applies to the lawn scene was also checked by computation, and the checks are listed at the end.

Throughout, a scene has m slots. At slot j there is a finite set L_j of live classes, and a class $k \in L_j$ has a weight $\mu_j(k)$, its number of takes. Adjacent slots are related by a compatibility matrix $M_j \in \{0,1\}^{L_j \times L_{j+1}}$, whose entry is one when the two classes may follow each other. An *admissible tuple* is a choice of one live class at each slot with every adjacent pair compatible.

A.1 Transfer Matrices

Proposition A.1 (Counting by matrix product)**.** *The number of admissible tuples is* $\mathbf{1}^{\top} M_1 M_2 \cdots M_{m-1} \mathbf{1}$*. The number of admissible take tuples, in which each class is weighted by its takes, is* $\mathbf{1}^{\top} D_1 M_1 D_2 M_2 \cdots M_{m-1} D_m \mathbf{1}$*, where* $D_j = \operatorname{diag}(\mu_j)$*.*

Proof. Let $v_1 = \mathbf{1}^{\top}$ and $v_{j+1} = v_j M_j$. We show by induction that $v_j(k)$ is the number of admissible prefixes of length j that end in the class k. For $j = 1$ every class is a prefix of length one. If the claim holds for j, then an admissible prefix of length $j+1$ ending in k' is an admissible prefix of length j ending in some k with $M_j(k,k') = 1$, followed by k', and distinct pairs give distinct prefixes. So $v_{j+1}(k') = \sum_k v_j(k) M_j(k,k')$, which is the k'-th entry of $v_j M_j$. The total is the sum of the entries of v_m. For the weighted count, each prefix carries the product of the weights of its classes, and the same induction applies to $w_1 = \mathbf{1}^{\top} D_1$ and $w_{j+1} = w_j M_j D_{j+1}$. □

When admissibility depends on a window of w consecutive classes, as in the rules of Chapter 5 whose frontier has width w, the same argument applies to a transfer matrix indexed by windows: a state is the last w classes, and a transition appends one class and

drops the first. The matrices are larger, with at most q^w rows for q classes at a slot, and the induction is unchanged. This is the sense in which the frontier width governs the cost.

A.2 Ranking and Unranking

Order the live classes at each slot, and order admissible tuples lexicographically. For a class k at slot j let $s_j(k)$ be the number of admissible completions from slot j onward that begin with k. Then $s_m(k) = 1$ and $s_j(k) = \sum_{k'} M_j(k, k') s_{j+1}(k')$, which is the vector $M_j M_{j+1} \cdots M_{m-1}\mathbf{1}$ computed from the right.

Proposition A.2 (Unranking). *Let T be the number of admissible tuples. The procedure that, given $0 \le r < T$, chooses at slot j the first class k, in order, that is compatible with the previous choice and satisfies $r < s_j(k)$ after subtracting from r the values $s_j(k')$ of every earlier compatible class k', and then continues with the reduced r, returns the rth admissible tuple in lexicographic order. It is a bijection from $\{0, \dots, T-1\}$ onto the admissible tuples, and its inverse sends a tuple t to $\sum_j \sum_{k' < t_j,\ k' \text{ compatible}} s_j(k')$.*

Proof. At slot j, with the prefix fixed, the admissible completions are partitioned by the class chosen, and the classes appear in order, so the completions through the first a compatible classes occupy exactly the positions below $\sum s_j(k')$ over those classes. The rth completion therefore lies in the block of the class at which the running sum first exceeds r, and its position inside the block is the remainder. By induction on the number of remaining slots, the procedure returns the rth completion of the block. The inverse formula adds the sizes of the blocks that precede the tuple at each slot, which is its position. Each map undoes the other, so both are bijections. □

The table of s_j is computed in $O(mq^2)$ additions for q classes at a slot, and each rank or unrank takes $O(mq)$ comparisons.

A.3 Hall's Theorem and Its Worldwide Form

Theorem A.3 (Hall). *Let X and Y be finite sets and $E \subseteq X \times Y$, and for $S \subseteq X$ let $N(S)$ be the set of elements of Y adjacent to some element of S. There is an injective map $f : X \to Y$ with $(x, f(x)) \in E$ for all x if and only if $|N(S)| \ge |S|$ for every $S \subseteq X$.*

Proof. If f exists, then f maps S injectively into $N(S)$, so $|N(S)| \ge |S|$. For the converse we use induction on $|X|$. If $|X| = 1$ the condition says that the single element has a neighbor. Suppose $|X| > 1$ and the condition holds. There are two cases.

First suppose $|N(S)| \ge |S| + 1$ for every nonempty proper subset S of X. Choose $x \in X$ and a neighbor y of x, which exists because $|N(\{x\})| \ge 1$. Delete x and y. For a nonempty $S \subseteq X \setminus \{x\}$, deleting y removes at most one neighbor, so its new neighborhood has at least $|N(S)| - 1 \ge |S|$ elements. By induction the rest has a matching, and adding (x, y) completes it.

Otherwise some nonempty proper subset S has $|N(S)| = |S|$. The condition holds in the graph restricted to S and $N(S)$, since the neighborhood of a subset of S lies in $N(S)$, so by induction S is matched into $N(S)$. In the graph on $X \setminus S$ and $Y \setminus N(S)$ the condition also holds: for $T \subseteq X \setminus S$ the new neighborhood is $N(T) \setminus N(S)$, whose size is $|N(T \cup S)| - |N(S)| \geq |T \cup S| - |S| = |T|$. By induction it has a matching, and the two matchings use disjoint elements of Y, so together they match X. □

Corollary A.4 (Worldwide feasibility)**.** *Let viewers be grouped by locale p with U_p viewers, each adjacent to exactly the set G_p of global classes that locale can realize. Pairwise distinct editions exist for all viewers if and only if for every set P of locales, $\left|\bigcup_{p \in P} G_p\right| \geq \sum_{p \in P} U_p$. If the sets G_p are pairwise disjoint, the condition reduces to $U_p \leq |G_p|$ for each locale.*

Proof. For a set S of viewers let $P(S)$ be the set of locales that S meets. Then $N(S) = \bigcup_{p \in P(S)} G_p$ depends on S only through $P(S)$, and for a fixed P the set S with the smallest $|N(S)| - |S|$ is the one containing all viewers of the locales of P, so it is the worst case. Hall's condition for all S is therefore the stated condition for all P. For disjoint G_p the union has size $\sum |G_p|$, and the condition for P follows from the conditions for its members. □

Corollary A.5 (Repair at a given radius)**.** *Let the viewers whose tuples are invalid be adjacent to the free valid tuples within Hamming distance r of their own. A repair that moves only invalid viewers, to distinct free valid tuples, at radius at most r, exists if and only if Hall's condition holds in this graph.*

This is Hall's theorem for the graph of the statement, and the least radius for which the condition holds is the quantity called r^* in Chapter 10.

A.4 Minimum-Cost Assignment

Let $n \leq N$, and let c_{ij} be real costs for $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, N\}$. An *assignment* is an injective map f from rows to columns, and its cost is $\sum_i c_{i,f(i)}$. In Chapter 10 the rows are the viewers who must move, the columns are the free valid tuples, and c_{ij} is the Hamming distance.

Proposition A.6 (Optimality certificate)**.** *Suppose numbers u_i and $v_j \leq 0$ satisfy $u_i + v_j \leq c_{ij}$ for all i, j. Then every assignment costs at least $\sum_i u_i + \sum_j v_j$. If an assignment f has $c_{i,f(i)} = u_i + v_{f(i)}$ for all i, and $v_j = 0$ for every column not used by f, then f is optimal.*

Proof. For any assignment g, $\sum_i c_{i,g(i)} \geq \sum_i (u_i + v_{g(i)}) = \sum_i u_i + \sum_{j \in g(\text{rows})} v_j \geq \sum_i u_i + \sum_j v_j$, the last step because the omitted v_j are nonpositive. For f as described the first inequality is an equality, and the sum over used columns equals the sum over all columns, since the unused v_j are zero. So f attains the bound. □

Theorem A.7 (Cycle criterion and existence of certificates)**.** *Let $n = N$ and let f be a bijection. Define on the rows the weights $W_{ii'} = c_{i,f(i')} - c_{i,f(i)}$. Then f is optimal if and only if the graph on*

the rows with these edge weights has no cycle of negative total weight, and in that case certificates u, v as in Proposition A.6 exist. The case $n < N$ reduces to this one by adding $N - n$ rows of zero cost to every column.

Proof. Suppose a cycle $i_1 \to i_2 \to \cdots \to i_k \to i_1$ has negative weight. Define g to agree with f off the cycle and to send i_s to $f(i_{s+1})$, indices modulo k. It is a bijection, and its cost differs from that of f by the weight of the cycle, so f is not optimal. Conversely suppose there is no negative cycle. Then shortest-path distances $d(i)$ from an added source joined to every row by an edge of weight zero are finite, and they satisfy $d(i') \le d(i) + W_{ii'}$ for every pair. Put $u_i = c_{i,f(i)} - d(i)$ and $v_{f(i)} = d(i)$. Then $u_i + v_{f(i)} = c_{i,f(i)}$, and for any i', $u_i + v_{f(i')} = c_{i,f(i)} - d(i) + d(i') \le c_{i,f(i)} + W_{ii'} = c_{i,f(i')}$, where the inequality is the shortest-path inequality $d(i') - d(i) \le W_{ii'}$, so u, v satisfy the inequalities. For $n = N$ every bijection g uses every column, so $\sum_i c_{i,g(i)} \ge \sum_i (u_i + v_{g(i)}) = \sum_i u_i + \sum_j v_j$ with no sign condition on v, and equality holds for f, which is therefore optimal. For $n < N$ add the dummy rows. An optimal assignment of the real rows, with the dummy rows placed on the unused columns, is an optimal bijection of the padded problem, since a dummy row costs nothing wherever it goes. The certificate of the padded problem has the sign condition of Proposition A.6 after one normalization. A dummy row k placed on an unused column j_k has $u_k + v_{j_k} = 0$ and $u_k + v_j \le 0$ for every column j, so $v_j \le v_{j_k}$ for all j. This holds for every dummy row, so all unused columns share one value v^*, and $v_j \le v^*$ for every column. Subtracting v^* from every v_j and adding it to every u_i of the real rows keeps every inequality and every equality, makes the unused columns zero and the used columns nonpositive, and leaves the sum unchanged. □

The theorem is the strong duality of the assignment problem, proved here through shortest paths. It does not give an algorithm, and the Hungarian method of Kuhn [16] finds an optimal assignment and a certificate in time cubic in the number of rows. The text uses a library routine for the assignment, and the check of the previous theorem, that no negative cycle remains, is the independent test of its output.

A.5 Packing and Covering

Let $\mathcal{P}$ be a finite set with a distance d, and let $\varepsilon \ge 0$ be an integer. A subset is an *ε-packing* if its points are pairwise more than ε apart, and its largest size is the packing number M_ε. A subset is an *ε-covering* if every point of $\mathcal{P}$ is within ε of one of its points, and its smallest size is the covering number N_ε.

Proposition A.8 (Packing and covering)**.** *For every $\varepsilon \ge 0$, $N_\varepsilon \le M_\varepsilon \le N_{\lfloor \varepsilon/2 \rfloor}$.*

Proof. A maximal ε-packing is an ε-covering, since a point farther than ε from all of its members could be added, so $N_\varepsilon \le M_\varepsilon$. For the other inequality take a $\lfloor \varepsilon/2 \rfloor$-covering and an ε-packing. Each packing point lies within $\lfloor \varepsilon/2 \rfloor$ of some covering point. Two packing points with the same covering point would be within $2\lfloor \varepsilon/2 \rfloor \le \varepsilon$ of each other by the triangle

inequality, contrary to the packing condition, so the assignment of packing points to covering points is injective, and $M_\varepsilon \le N_{\lfloor \varepsilon/2 \rfloor}$. □

Proposition A.9 (Singleton bound)**.** *Let $\mathcal{P} \subseteq A_1 \times \cdots \times A_m$ with the Hamming distance, and let $q = \max |A_j|$. An ε-packing has at most $q^{m-\varepsilon}$ points whenever $\varepsilon < m$.*

Proof. Two points of an ε-packing differ in at least $\varepsilon + 1$ coordinates. Delete the same ε coordinates from every point. Two points that agreed on the remaining $m - \varepsilon$ coordinates would differ in at most ε coordinates, so the restricted points are distinct, and there are at most $q^{m-\varepsilon}$ restrictions. □

The bound is attained when there is a code with those parameters. For $m \le q + 1$ and q a prime power, the Reed–Solomon codes of Reed and Solomon [17] attain it for every $\varepsilon < m$, and the text uses them without reproducing the construction. The two smallest cases that the text mentions are within reach of direct computation, and the largest ternary codes of length four with distance two and three have 27 and 9 words, which equal 3^{4-1} and 3^{4-2}.

A.6 What Was Checked

The script `verify_appA.py` checks the statements of this appendix by computation. It confirms Proposition A.1 against enumeration for all six pools of the lawn scene, with counts of 494, 238, 320, 96, 2 and 1 and take-tuple counts of 120,042, 57,834, 10,240, 3,072, 204,800,000 and 102,400,000. It checks Hall's condition against a maximum matching on 3,000 random bipartite graphs, of which 1,177 satisfied the condition, with no disagreement. It checks the output of the assignment routine against brute force on 400 random matrices of side two to six, and finds no negative cycle at the routine's optimum in any. It checks the inequalities of Proposition A.8 on 120 random subsets of a Hamming space, with no violation. And it solves exactly the largest ternary codes of length four with distance two and three, which have 27 and 9 words, the Singleton values. These checks are evidence about the implementations and the instances, and the proofs above are what establish the statements.

Appendix B

Verification Scripts

Each chapter that states a count has a script that reproduces it, and this appendix says what the scripts are, what they were run on, and what running them does and does not establish. It states the instances plainly, because every count in the book is a count on a particular instance and the instance is part of the result.

B.1 Running the Scripts

The scripts are Python programs written for Python 3 with the packages numpy, scipy, and, for the two scripts that read the measured data, pandas. They were run under Python 3.13 with numpy 2.5, scipy 1.18, and pandas 3.0. Thirteen of them import `lawnmodel.py`, the shared model of the lawn scene, and so must be run from the folder that contains it and without Python's isolated-mode flag; the six that are self-contained are the scripts for Chapters 1, 3, 4, 6, and 14 and the supplementary script for Chapter 17. The scripts for Chapters 6 and 17 read the per-second table and the block list in the folder `data`, and take their paths relative to the folder they are run from. The script `run_all.sh` runs every script in order, writes each output to a log, and reports the exit status and the time. Its variable `SKIP` leaves chosen scripts out, which is useful because four of them take more than half a minute.

```
./run_all.sh
SKIP="verify_ch02 verify_ch09 verify_ch10 verify_ch11" ./run_all.sh
```

Chapter 18 and Appendix C contain no counts and have no script, and the script `verify_ch17_shared.py` supplements `verify_ch17.py` with the figures on shared discussion in Chapter 17. Appendix A has one, `verify_appA.py`, which checks the statements of that appendix.

B.2 The Scripts and Their Instances

Table B.1 lists each script with the instance it computes on, its sources of randomness, and the seconds it took on the machine used. Seeds are fixed in the code, so a rerun gives the same numbers. Times are for one run on one machine and are a guide only.

Table B.1: The verification scripts, their instances, and their running times.

Script	Instance and randomness	Solver and data	Seconds
`verify_ch01`	the three lawn productions A, B, C with their own definition of the pools, the script, the five slots, and the audiences 60, 30, 2; seeds 2026 + ord of the production letter, and 7	none	1
`verify_ch02`	the corrected pools of A and B; Hamming dissimilarity on class tuples; tolerances 0 to 2	integer program, time-limited	235
`verify_ch03`	the engine scene of 22 histories, 126 class tuples; no randomness	none	0
`verify_ch04`	the lawn scene with its own pools, frontier tables and unranking; seeds 11 and 3	none	10
`verify_ch05`	the lawn scene with interval-closure rules and synthetic scenes of varying width; no randomness	none	0
`verify_ch06`	the per-second table of the measured film, 5,222 rows, and the block list	`data/` files	1
`verify_ch07`	claims on the lawn scene, with allocations drawn at seeds 2026 + ord and 7000 + ord	none	1
`verify_ch08`	order paths of the three amendments for A and B; seed 2026 + ord	none	0
`verify_ch09`	capacity, robust and invariant capacity of the lawn pools; exhaustive enumeration	none	33

Script	Instance and randomness	Solver and data	Seconds
`verify_ch10`	repair of the allocations of A, B, C; seed 2026 + ord, and explicit seeds in the witness search	assignment routine	34
`verify_ch11`	two locales with the pool of A after the agreement rule; codes over three, four and eight symbols	integer program, time-limited	33
`verify_ch12`	allocations at seed 2026 + ord, takes at 7000 + ord, a draw at 99	assignment routine	1
`verify_ch13`	viewing budget with the measured film length of 5,222 s; the corrected pools; packing numbers of Chapter 2 entered by hand; synthetic scenes of random compatibility, seed 1	none	0
`verify_ch14`	the engine scene and later films defined by what they commit to; no randomness	none	0
`verify_ch15`	the corrected pools; random corpora, seeds 15 and 150	integer program, time-limited	9
`verify_ch16`	the lawn pools; 500 random orders for greedy packing, seed 16	none	2
`verify_ch17`	the engine scene with three locales of thirty, seed 17; a two-scene film, seed 1700, with an invented dishwashing scene; the measured passage, seed 5 for take draws	assignment routine; `data/` files	1
`verify_ch17_shared`	the engine scene with its 32 faithful editions and the pairs of them; no randomness	none	0
`verify_appA`	the six lawn pools; 3,000 random bipartite graphs, 400 cost matrices, a Hamming-space packing, seed 3	integer program, 120 s limit	2

Three instances need a word. The lawn scene's pools are defined once in `lawnmodel.py` and separately inside the scripts for Chapters 1 and 4, so those two chapters are computed independently of the model that the later scripts share, and the two definitions give the same counts: 494 editions of A at the premiere and 238 after the agreement rule. The dishwashing scene of Chapter 17 is defined only in `verify_ch17.py`, with pools that were invented for

that study. And the measured film is represented only by the table of per-second levels and the list of occupation blocks that were produced by the analysis of Chapter 6; the soundtrack itself is not part of the distribution.

B.3 What a Run Establishes

A run establishes that the text's counts follow from the instance as the script defines it. It does not establish more. The scripts that import the shared model share its errors, if it has any, and the agreement of the independent definitions in Chapters 1 and 4 with the shared model is the only cross-check of the pools. Many scripts print their results for comparison with the text and do not assert them. Some contain assertions, such as the checks on the number of blocks and windows in Chapter 6, but most counts are checked only by comparing what is printed with what is written. The script `check_counts.py` adds a regression check for the headline counts of the quick scripts: it runs them and fails if a count that the text states is missing from the output. It covers a fraction of the counts, and a pass says that those counts still match the text. A disagreement between a script and the text indicates an error in one of them, and it does not say which.

Two further limits apply to the solver-based scripts. Where an integer program is solved within a time limit, the result is either optimal or a bound, and the scripts print which. In Chapter 2 the covering number of production A at tolerance one is reported as a range from 25 to 31 because the solver did not close the gap in its limit, and it was not closed by any later run, which is one of the open problems of Chapter 18. On a slower machine a result reported as optimal here might be reported as a bound. And the random draws are single seeds or a stated number of draws, so the counts that depend on them, such as the anonymity sets of Chapter 12 and the survivors of Chapter 15, are values for those draws and would differ in others, though the structure would not.

B.4 What the Appendix Assumes

The appendix describes the scripts as they stood when it was written. Their running times and package versions are those of one machine on one day, and a later change to any script or to a package could change a result without changing this text. The sources of the data of Chapter 6 are the analysis cited there, and the appendix does not distribute the film or its soundtrack.

Appendix C

Notation and Relation to the Earlier Volume

This appendix does three things. It tabulates the notation of the book, including the places where one symbol carries two meanings. It sets out the correspondence between the objects of the earlier volume, *Generative Film: Categories, Sheaves, and Types for Cinema* [1], and the objects of this one. And it lists the points at which the two books use different words for the same thing, or the same word for different things. Where the correspondence is a reading and not a theorem, the appendix says so.

C.1 Notation of This Book

The table gives the symbols in order of first use. The chapter column gives where the symbol is introduced or most used.

Symbol	Meaning	Chapter
h^*	the script history of a scene	1, 3
m	number of slots in a scene plan	1
t_j, V_j	story time and set of strips in frame at slot j	1
r_j	required appearance at slot j, mown or unmown for each strip in frame	1
$F(s)$	footprint of strip s, the set of slots that show it	1
A, B, C	the three productions of the lawn scene, with audiences 60, 30 and 2	1
D	the direction rule on adjacent slots	1, 4
G, $\alpha_a, \alpha_b, \alpha_c$	the agreement rule, and its restrictions to the strips a, b, c	8, 9
ω	the take-level amendment withdrawing a take variant	9

Symbol	Meaning	Chapter
K	number of distinguishable editions; in Chapter 14 a viewer's knowledge	1, 14
B	storage cost of footage, rules and metadata; also the second production in tables	1
ρ	the rate $\log_2 K/B$; in Chapter 15 a neighborhood radius	1, 15
ε	tolerance of perceptual equivalence	2
M_ε, N_ε	packing and covering numbers at tolerance ε	2, A
π, Cap	projection of realizations to class tuples; capacity $\lvert\pi(X)\rvert$	9
Rob, Inv	robust and invariant capacity of a family of amendments	9
U, U_p	size of an audience; of the audience of locale p	1, 11
r^*	least radius of a repair	10
G_p	set of global classes that locale p can realize	11, A
ℓ, s	number of slots with a local trace, and of shared slots, in Chapter 11	11
N, w, h, y	viewings, hours per week, hours per viewing, years, in Chapter 13	13
H, C_f	set of histories of the engine scene; the set a film f commits to	3, 14
$R(K)$	the readings of the histories in a knowledge state K	14
X	a reference corpus in Chapter 15; a set of realizations in Chapter 9	9, 15
M_j, D_j	compatibility matrix and weight matrix of slot j	A
$s_j(k)$	number of admissible completions from slot j beginning with class k	A

Five of these symbols are reused. The letter ρ is the rate in Chapter 1 and the neighborhood radius in Chapter 15, and the chapters do not meet. The letter K counts editions in Chapter 1 and names a knowledge state in Chapter 14. The letter N is the number of viewings in Chapter 13, and the covering number N_ε carries a subscript. The letter G is the agreement rule in Chapters 7 to 10 and the set G_p of global classes of a locale in Chapter 11, the latter always subscripted. The letter M is the packing number when subscripted by a tolerance and the compatibility matrix of a slot in Appendix A when subscripted by a slot. The letter B is a storage cost in Chapter 1 and a production only inside the tables of the lawn scene. None of these meanings is used in the same passage as another.

C.2 Correspondence of Objects

The earlier volume treats film as a composition problem and develops category, sheaf, and type theory for it. This book is about a different question, how many variants of a film a production can sustain, and it uses little of the earlier machinery directly. The correspondences are of three kinds: an object of this book is an instance of an object of the earlier volume, a result of this book can be read as a result of the earlier one, or the two stand in a looser relation. The table says which.

Earlier volume	This book	Relation
Layered world: history $A \subseteq E^*$, fold $M : A \to Q$, appearance $R : Q \to I$ (Chapter 11)	Script history h^*, set of mown strips $M(h^*_{\le t})$, picture of the strips in frame at a slot	Instance. The lawn scene is a layered world whose state is the set of mown strips.
Authority principle (Principle 11.5): history authoritative at commitment, appearance never authoritative	Ledger and archive of Chapter 12: records of assignments and takes are kept, rendered frames are not	Reading. The ledger stores the tuple from which the appearance is recomputed.
Future equivalence $\sim$ and the quotient $A/\!\sim$, the minimal state space (Proposition 11.2)	Future-equivalence quotient used to count overcapture by noise in Chapter 3	Instance.
Several folds of one history: physical, semantic, provenance (Section 11.3)	Commitment set of an edition, readings of Chapter 3	Reading. A commitment set is a semantic fold of the histories that agree with what an edition shows.
Admissible histories, prefix-closed A (Chapter 9, 11, 12)	Admissible class tuples under the direction and agreement rules	Analogy, one level up. Admissibility here is of tuples of slot classes, not of event words.
Viewer classes V, schedule $\mathbb{N} = \bigsqcup N_v$, rate budget $\lvert V\rvert \le \lfloor r/r_{\min} \rfloor$ (Proposition 13.1)	Locales and audiences, Hall condition (Chapter 11, Appendix A)	Different resource. The earlier budget shares frames of one screen, and this book counts editions and says nothing about how they are delivered.

Earlier volume	This book	Relation
Joint constraint $J \subseteq \prod_v F_v$, subsheaf if and only if a product (Proposition 13.2)	Worldwide distinctness of editions, a non-product constraint on the editions of all viewers	Reading. Each viewer's edition can be admissible and the tuple of editions not, which is the failure of Hall's condition in Chapter 1.
Branching film, tree T with L leaves, shared segments (Section 13.3, Principle 13.4)	Shared slots among locales (Chapter 11)	Reading. A shared slot cannot carry a local trace, as a shared segment cannot require a witness from one branch.
Version, situation θ, dissimilarity d_θ, three response variables (Section 14.1)	Edition, the detection, comprehension and affect dissimilarities (Chapter 2)	Instance. The situation θ is left unspecified in this book.
Thresholded similarity $\approx_\varepsilon$ is not transitive (Proposition 14.3), drift $\leq n\varepsilon$	Non-transitivity of ε-similarity, packing and covering numbers (Chapter 2)	Instance, with a counting form added.
Effective dimension d_{eff} from singular values of a Jacobian (Proposition 14.6)	Packing number M_ε and covering number N_ε	Counterpart. d_{eff} counts continuous directions, and the packing number counts distinguishable editions of a discrete pool.
Constraints, theory and model operators Th, Mod; adding constraints shrinks Mod (Chapter 5, Exercise 5.2)	Amendments as subsets $X' \subseteq X$ of realizations (Chapters 8 and 9)	Instance. A correction is a constraint added to the script.
Compression as adjunction $D \dashv G$ (Section 5.3)	The rate $\rho = \log_2 K/B$ (Chapter 1)	Loose. The rate is a number and not an adjunction. B plays the role of the description.
Command language $\mathcal{K}$, realizations $R(c)$, refinement (Definition 16.3)	Nested commands narrowing slot domains (Chapter 16)	Instance for the narrowing. Prefix-free decoding is not used.

Earlier volume	This book	Relation
Acoustic occupation, contracts, the measured passage (Sections 15.3 and 17.9)	Chapter 6 and Study Three of Chapter 17, with the same data	Same data. This book adds timing editions and a correction to the reference.
Register of Claims (Appendix D)	Statements of assumptions beside each result, and Chapter 18	Different organization. This book has no register, and collects its open questions instead.

The relations should be read with care. Only the rows marked as instances, which are the layered world, the future-equivalence quotient, the perceptual dissimilarities, the amendments, the commands and the measured passage, claim that an object of this book *is* an object of the earlier one. The rows marked as readings are interpretations that this book offers and does not prove. The row on viewer classes is marked as a different resource because the two books count different things: the earlier one a rate of frames, and this one the number of distinct editions.

C.3 Different Words for the Same Thing

The working title of this volume was *A Cinema of Individual Editions*, and the standalone chapter files retain it in their headers; the merged book carries the title *Hypernormal Cinema*.

The words below differ between the two books while the objects are the same or nearly so.

Earlier volume	This book	Note
Version	Edition	The earlier word attaches presentation conditions to a film, and this book uses the same idea for the film that a viewer is shown.
Viewer class	Audience, locale	The earlier word names a selector worn by viewers, and this book groups viewers by production or locale.
Appearance	Take shown at a slot, class at a slot	The earlier word is a function of state, and this book’s classes are the equivalence classes of takes under exact slotwise equivalence.

Earlier volume	This book	Note
Scenario space, film as a path	Space of possible films, edition as a path through the graph of editions	Chapter 4 draws the graph of editions of a scene, a path being a choice of class at each slot.
Effective dimension	Packing number, capacity	See the table of correspondences.
Tolerance ε and $\approx_\varepsilon$	Tolerance ε	Same symbol, with the books agreeing that thresholded similarity is not an equivalence.
Schedule	Cyclic schedule	The earlier word is a partition of frame indices, and this book's is an assignment of ranks to viewers over rounds in Chapter 13.

Three words are used differently. The word *admissible* describes event words in the earlier volume and tuples of classes here. The word *rate* is a number of frames per second there and the ratio $\log_2 K/B$ here. The word *principle* labels each modeling hypothesis in the earlier volume and is not used in this one, where hypotheses are stated in the passage where they enter and collected in Chapter 18. The word *fold* has the same meaning in both, a function from histories to what a history leaves behind, though this book uses it for a commitment set and not for a state.

C.4 What the Appendix Establishes and What It Assumes

The tables record the author's account of the two books and have not been checked against anything but the books. The notation table covers the symbols that a reader needs to follow the chapters and does not claim to be complete. The correspondence table is selective: the earlier volume's categories, sheaves, and dependent and linear types are not used in this book beyond the readings in the table, and a reader who wants the connection to those theories will find it in the earlier volume and not here.

Bibliography

[1] Flyxion. *Generative Film: Categories, Sheaves, and Types for Cinema*. Monograph, October 2026. `https://standardgalactic.github.io/kitbash/generative-film/generative-film.pdf`

[2] David Bordwell, Janet Staiger, and Kristin Thompson. *The Classical Hollywood Cinema: Film Style and Mode of Production to 1960*. Columbia University Press, New York, 1985.

[3] Richard P. Stanley. *Enumerative Combinatorics, Volume 1*, second edition. Cambridge University Press, Cambridge, 2012.

[4] Philip Hall. On Representatives of Subsets. *Journal of the London Mathematical Society*, 10(1):26–30, 1935.

[5] Hugo Krawczyk, Mihir Bellare, and Ran Canetti. HMAC: Keyed-Hashing for Message Authentication. RFC 2104, Internet Engineering Task Force, February 1997.

[6] John Black and Phillip Rogaway. Ciphers with Arbitrary Finite Domains. In *Topics in Cryptology, CT-RSA 2002*, Lecture Notes in Computer Science 2271, pages 114–130. Springer, Berlin, 2002.

[7] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*, third edition. Pearson, 2006.

[8] Nancy G. Kinnersley. The Vertex Separation Number of a Graph Equals Its Path-Width. *Information Processing Letters*, 42(6):345–350, 1992.

[9] Rina Dechter. Bucket Elimination: A Unifying Framework for Reasoning. *Artificial Intelligence*, 113(1–2):41–85, 1999.

[10] Neil Robertson and Paul D. Seymour. Graph Minors. III. Planar Tree-Width. *Journal of Combinatorial Theory, Series B*, 36(1):49–64, 1984.

[11] Stefan Arnborg, Derek G. Corneil, and Andrzej Proskurowski. Complexity of Finding Embeddings in a k-Tree. *SIAM Journal on Algebraic Discrete Methods*, 8(2):277–284, 1987.

[12] Hans L. Bodlaender. A Linear-Time Algorithm for Finding Tree-Decompositions of Small Treewidth. *SIAM Journal on Computing*, 25(6):1305–1317, 1996.

[13] J. Scott Provan and Michael O. Ball. The Complexity of Counting Cuts and of Computing the Probability That a Graph Is Connected. *SIAM Journal on Computing*, 12(4):777–788, 1983.

[14] Flyxion. *Acoustic Occupation: Salience, Recovery, and the Temporal Organization of Film Sound*. Unpublished monograph, September 2026.

[15] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, San Francisco, 1979.

[16] Harold W. Kuhn. The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955.

[17] Irving S. Reed and Gustave Solomon. Polynomial Codes over Certain Finite Fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960.

[18] Alexei Yurchak. *Everything Was Forever, Until It Was No More: The Last Soviet Generation*. Princeton University Press, 2006.

[19] Adam Curtis. *HyperNormalisation*. Documentary film, BBC, 2016.