Computational Complexity of Arranging Music

William Moses and Erik Demaine
MIT

$$\neg x_1 \vee x_2 \vee x_3$$

Domain & Motivation

Why is this problem interesting?

Arrangement

- Transcribe a piece written for instruments A to be played on instruments B

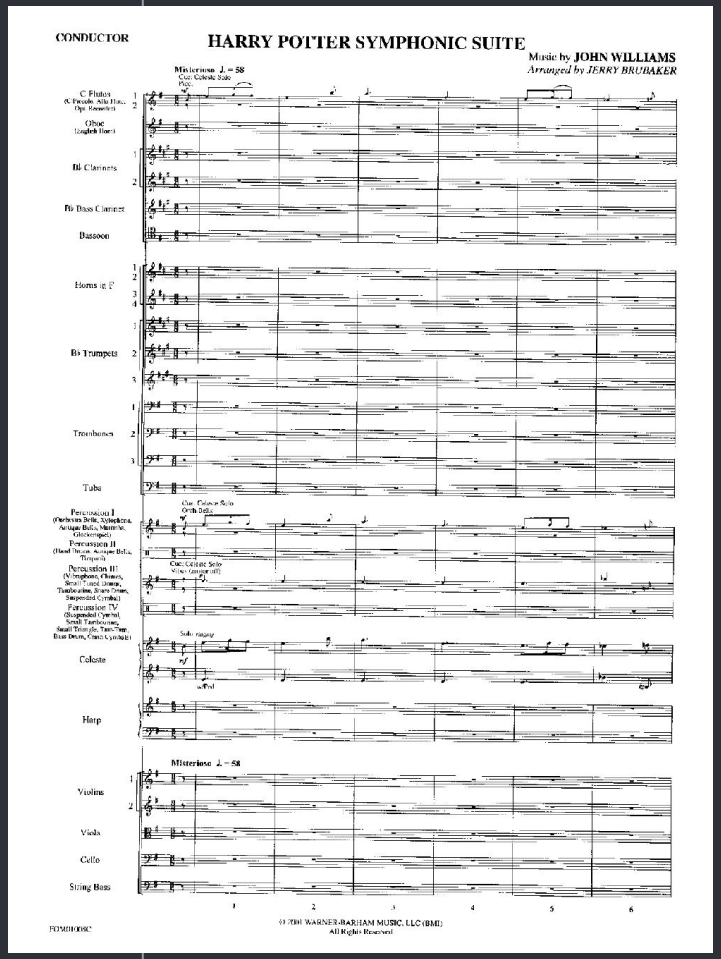
- Expand repertoire (Zelda Theme played by NSO, Canon in D on Guitar)
- Automated arrangement software

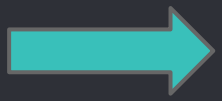
Musical Choreography

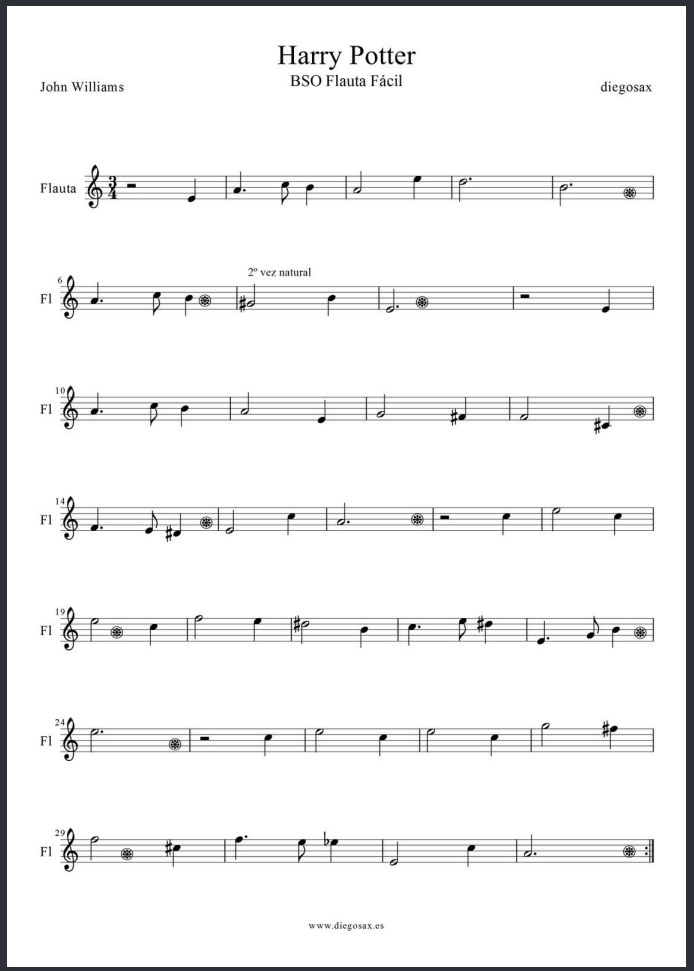
- Create a performance which fits with music
- Examples: Dancing, Cinema, Skating
- Similar to arrangement, representing moves by performer as notes on instrument

Problem Statement

Given a score consisting of *n* instrumental parts, does there exist a valid arrangement of the piece for one instrument?



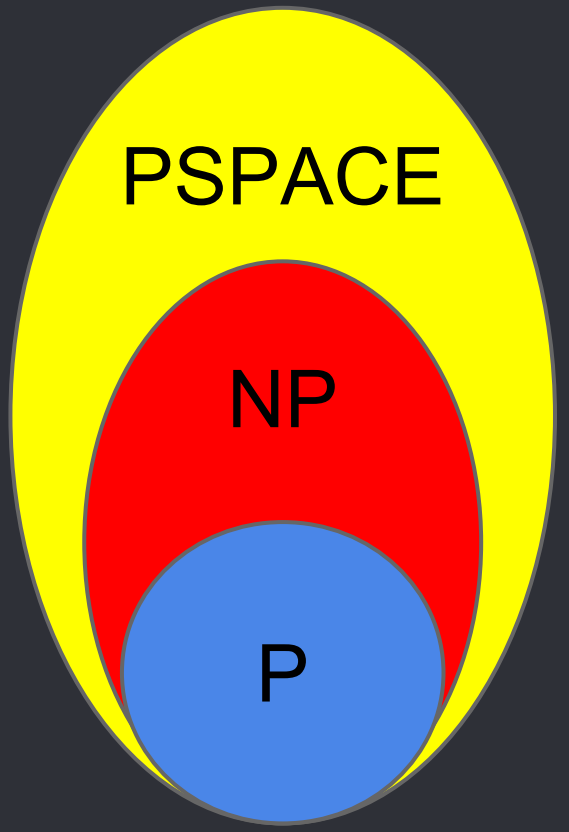




P vs NP (vs PSPACE …)

Computational Complexity

- Decision Problems
- How long will it take to compute?
- P := Solvable in Polynomial time
- NP := Nondeterministic Polynomial Time



Computational Complexity

- Reduction from A to B encodes A in B, means B at least as hard as A
- All problems in NP can be ‘reduced’ or encoded inside 3SAT

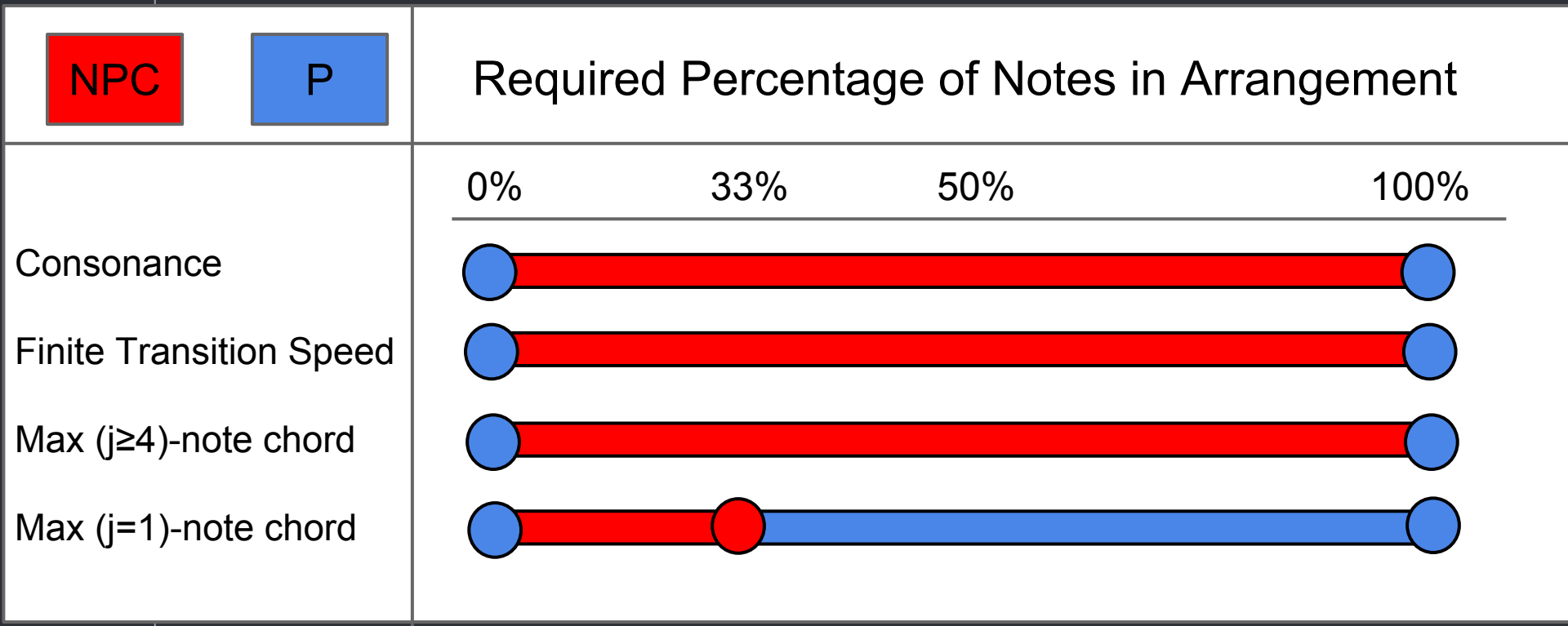
3SAT

- Does boolean formula have solution?
- Literal: Variable or not a variable
- Clause: 3 literals or'd together
- Formula: Series of clauses and'd together

$$\underbrace{(\neg x_1}_{\text{Literal}} \vee \underbrace{x_3}_{\text{Literal}} \vee \underbrace{x_4)}_{\text{Literal}} \wedge \underbrace{(x_2 \vee \neg x_3 \vee x_4)}_{\text{Clause}}$$

Summary Of Results

Summary of Results



Quantifying a “good” arrangement

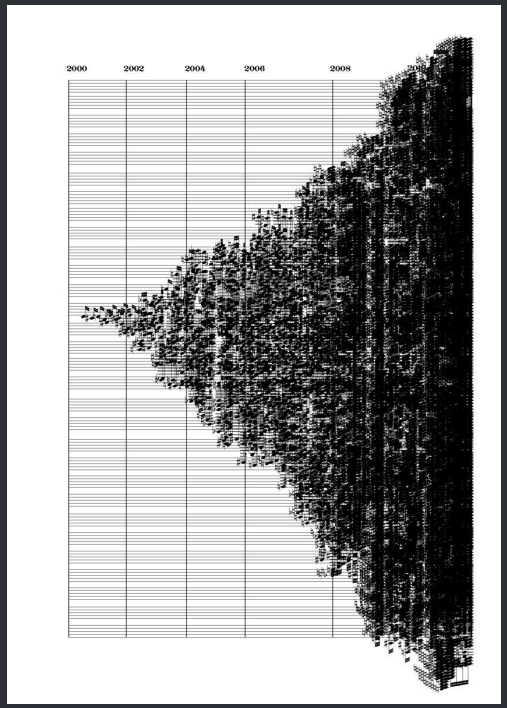
How do we deem an arrangement acceptable?

Criteria for Valid Arrangement

- Must be possible to be played / performed
- Must reflect the original intent of the piece (recognizable)
- Must be pleasing to listen to / watch

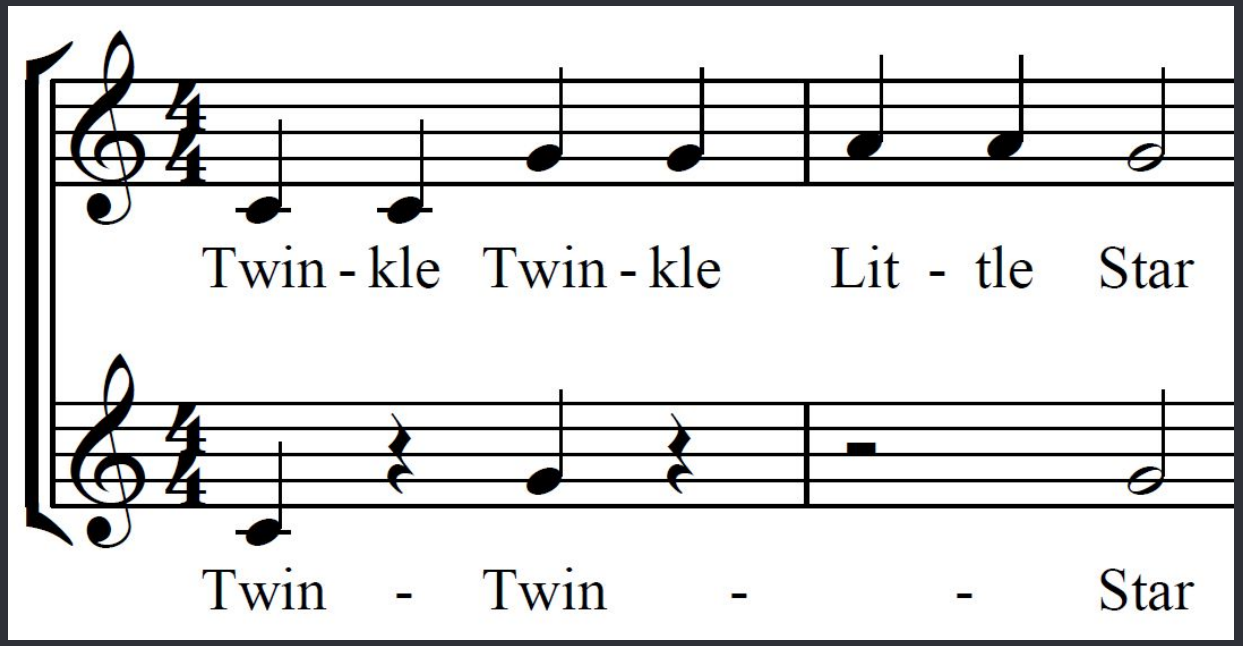
Limitations on performance

- Transition speed
- Number of simultaneous notes / actions



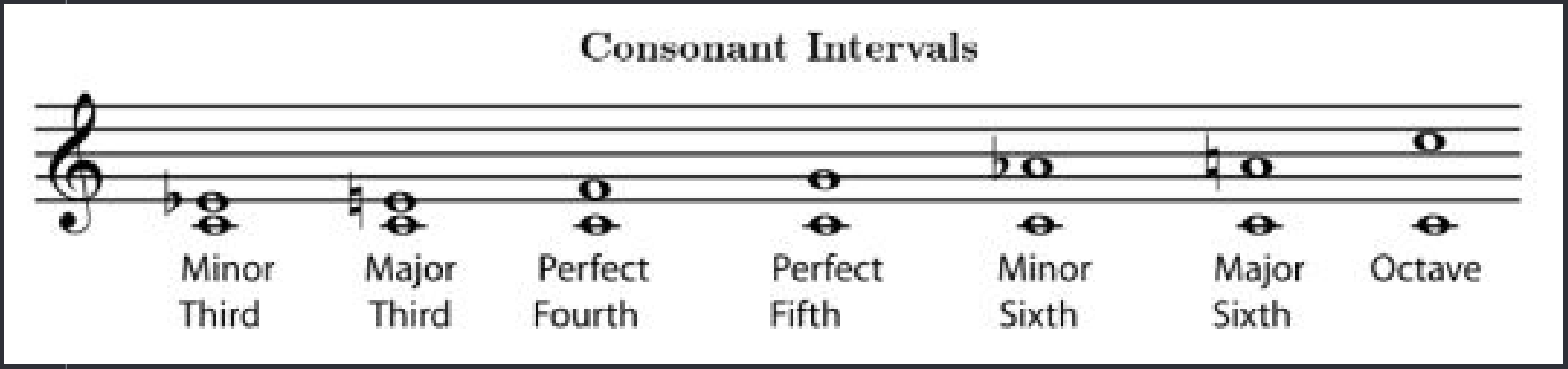
Original Intent / Recognizable

- Must maintain entire melodies
- Must keep certain percentage of original notes



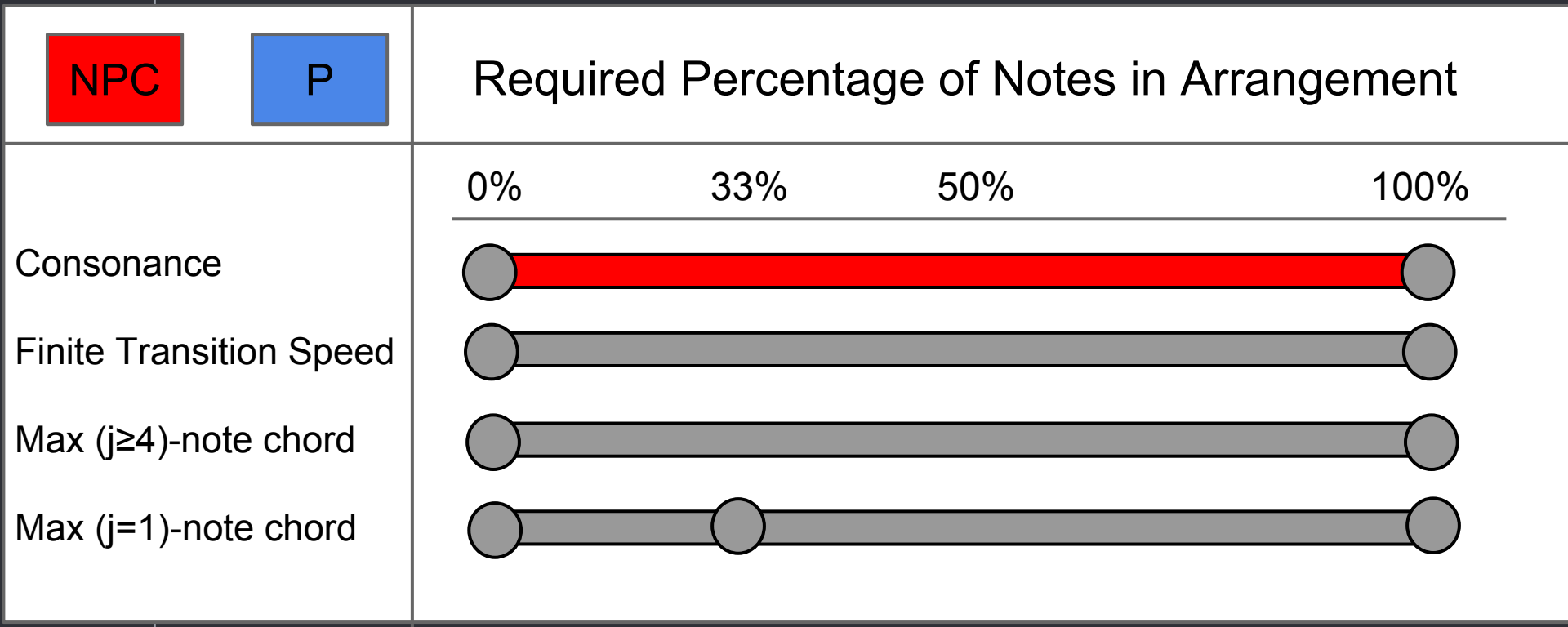
Pleasant Sounding

- Consonance: Simultaneous notes (chords) allowed in certain intervals



Hardness of Consonance

Summary of Results

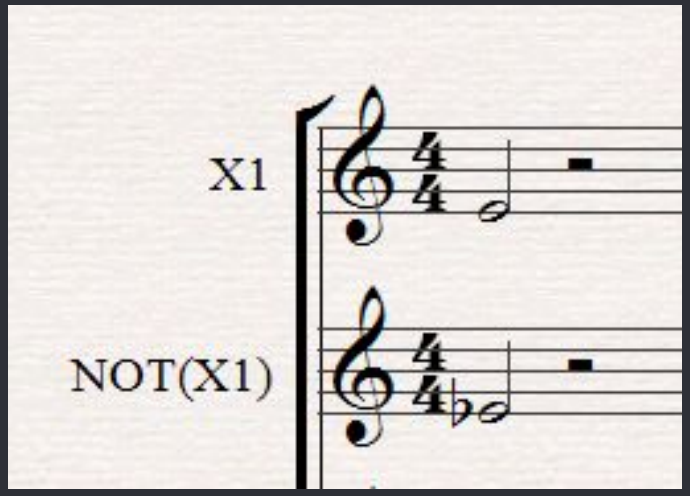


Consonance Requirements

- Parts are included or excluded in entirety (recognizable melody)
- Any simultaneous notes must be in consonance (pleasant sounding)
- At any given time, at least *n*% ($0 < n < 100$) of notes in the original song must be played (original intent)

Variable Gadgets

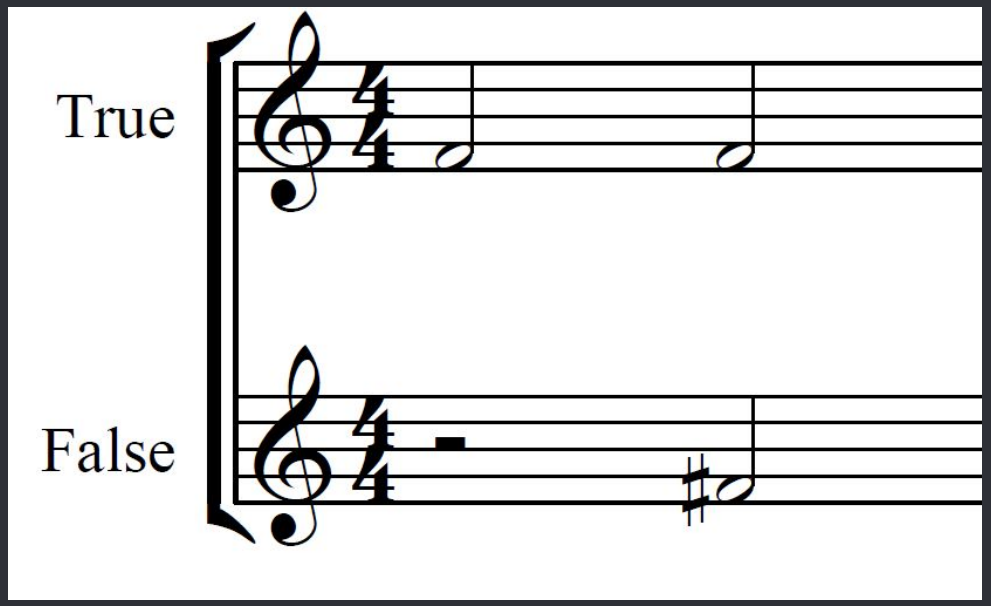
- Variables represented by the choice of one of two parts



- At most one part can be played from pleasant sounding requirement
- At least one part must be played from original intent requirement

True / False Literal

- We create parts which must be played (true) and must be omitted (false)

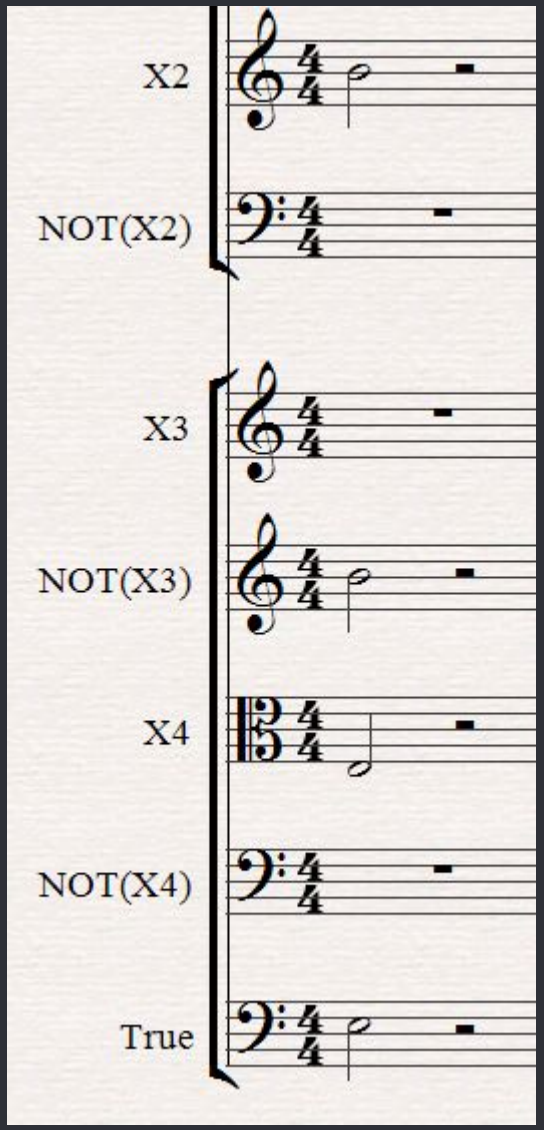


- A true literal can be created by simply having a note on its own -- which must be played in the arrangement

- A false literal can be created by creating a measure where it is in dissonance with a true literal

Clause Gadgets

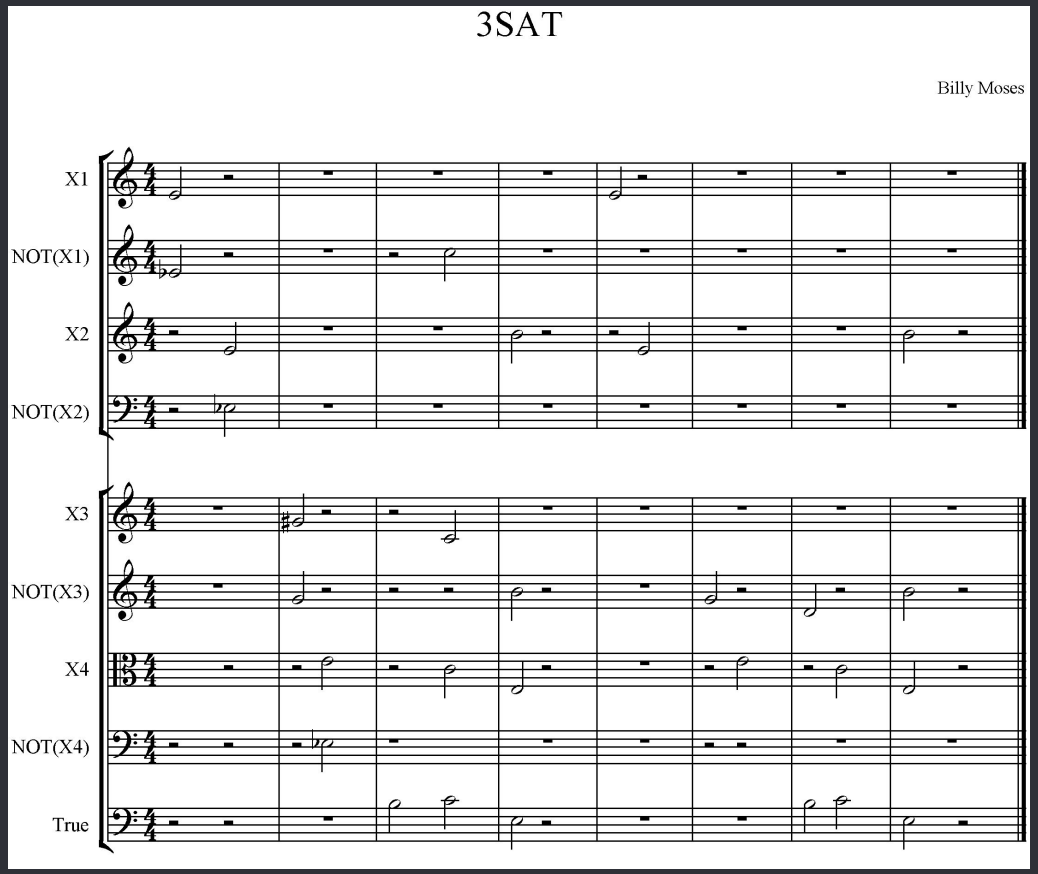
- Clauses represented by a measure three variable parts and some true/false literals
- Sufficient true / false literals added to ensure that n% of notes being placed requires at least one variable to be played (50% depicted)



Entire 3SAT

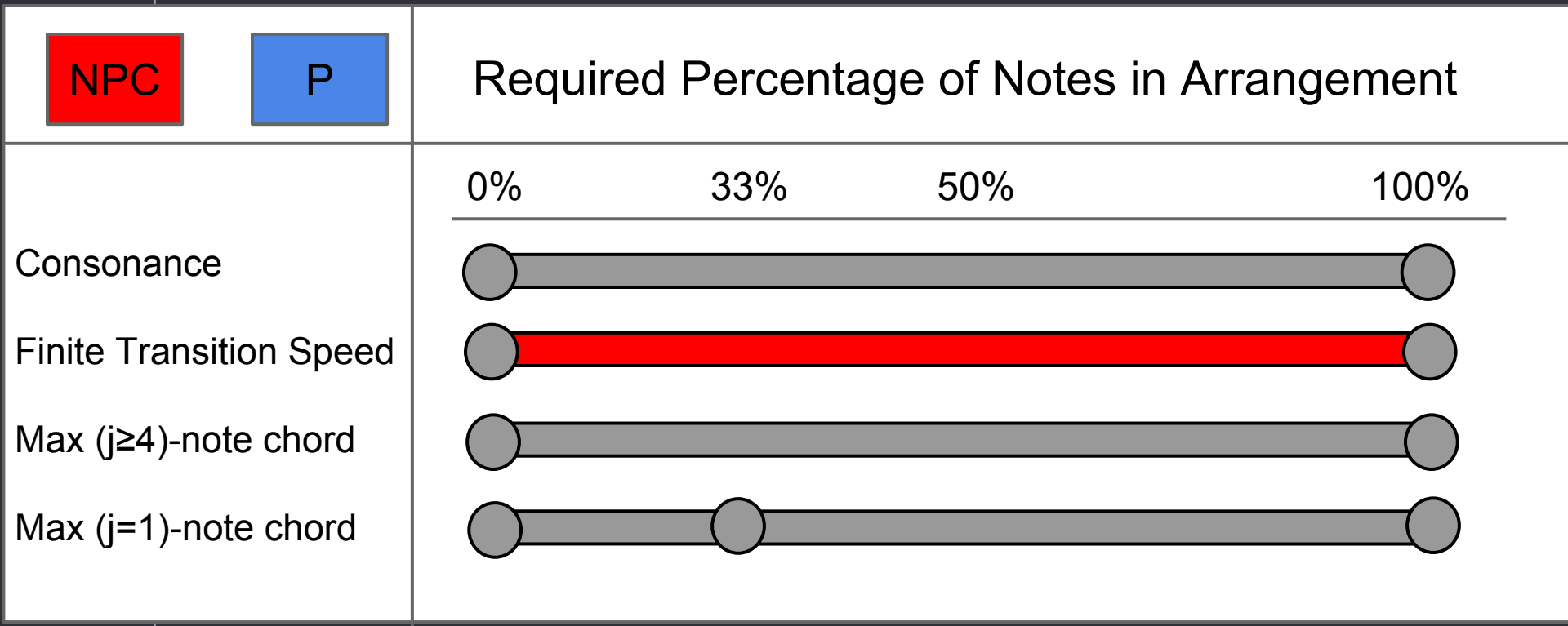
$$(\neg x_1 \vee x_3 \vee x_4) \wedge (x_2 \vee \neg x_3 \vee x_4)$$

- 3SAT represented as a song, followed by a satisfying assignment



Finite Transition Speed

Summary of Results

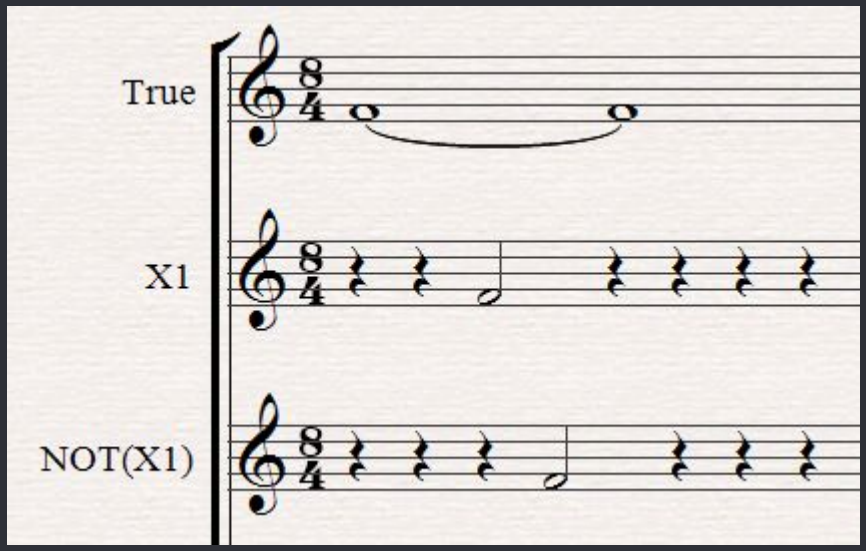


Transition Requirements

- Parts are included or excluded in entirety (recognizable melody)
- Notes or chords cannot change more frequently than a half note (playable)
- At any given time, at least *n*% (0 < n < 100) of notes in the original song must be played (original intent)

Variable Gadgets

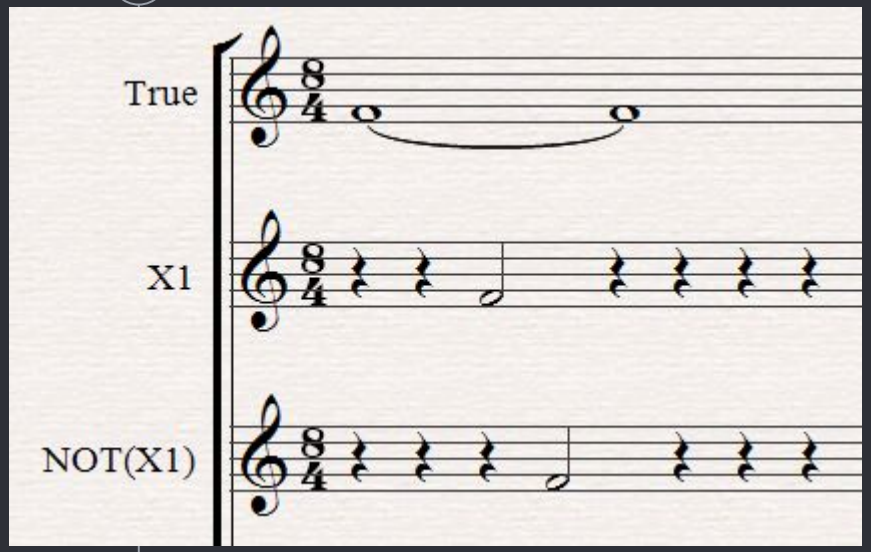
- Again variables represented by the choice of one of two parts

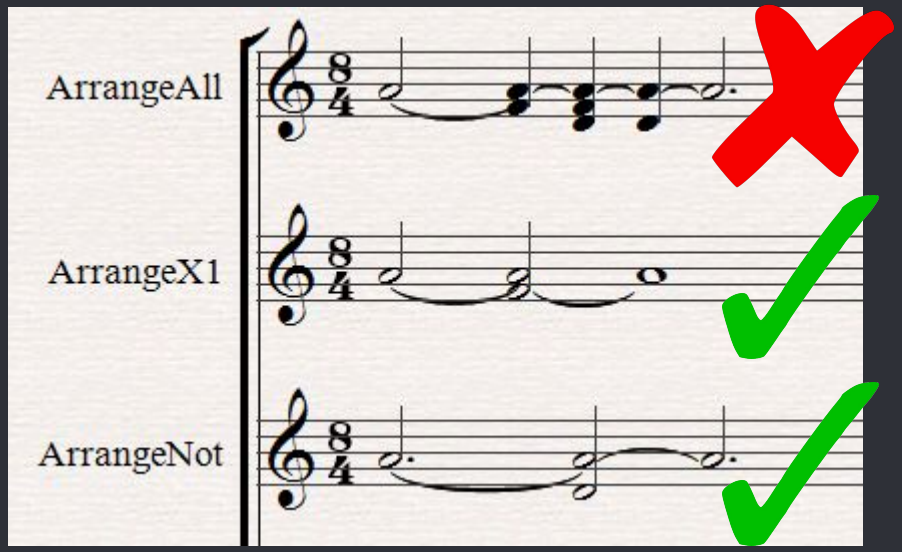


- By having the two parts to play notes offset by a quarter note, only one can be played without violating transition requirements.

- True / false literals used as padding to ensure at least n% notes played at any time

Variable Gadgets





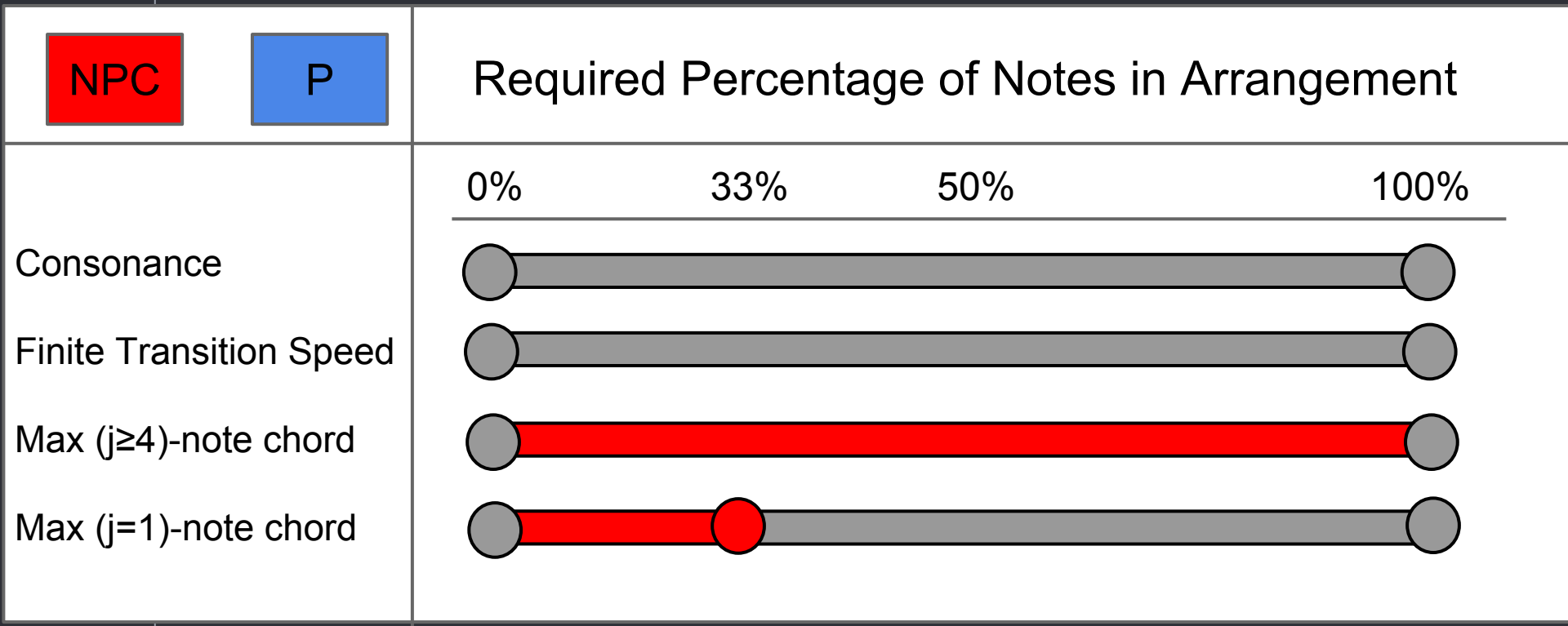
- Three arrangements of the 3SAT variable selection, selecting all parts; true and X1; and true and NOT(X1).

Transition Conclusions

- Using the transition exclusion established in the variable gadget, we can create clauses.
- Just like in the consonance problem these clauses can be used to create any 3SAT, showing the transition problem to be NP-hard

Max j-note Chord

Summary of Results

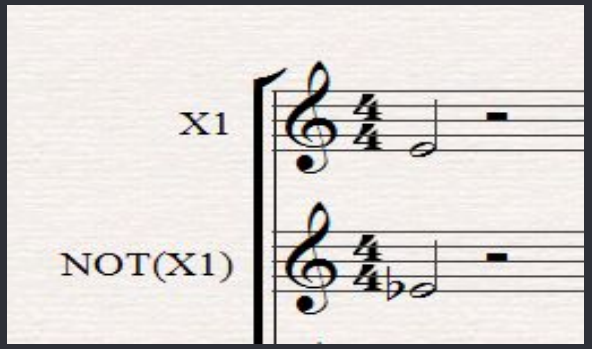


Max j-note Chord

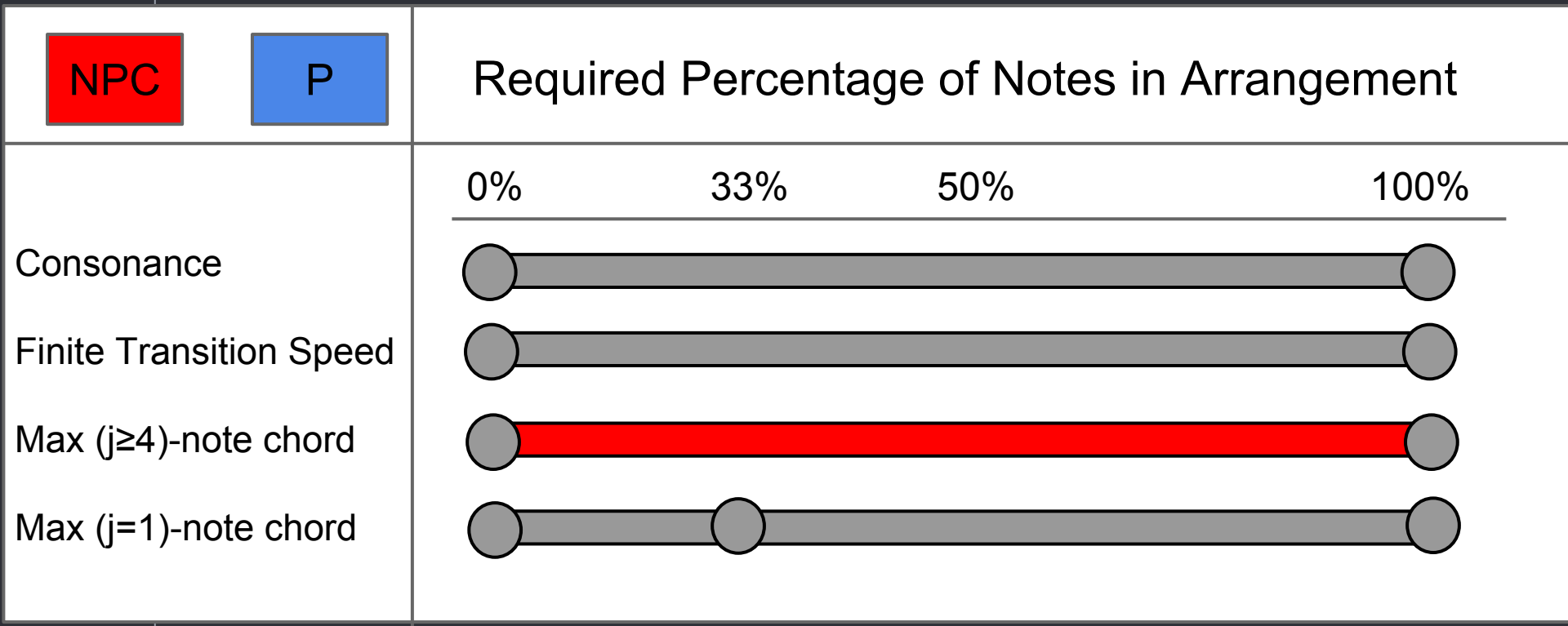
- Parts are included or excluded in entirety (recognizable melody)
- Only up to j notes can be played simultaneously (playable)
- At any given time, at least *n*% (0 < n < 100) of notes in the original song must be played (original intent)
- Using reduction from X-3SAT (only 1 variable is true)

Variable Gadgets (j >= 1)

- Again variables represented by the choice of one of two parts
- For j = 1, the same variable gadget from consonance can be used (since only one can be played).
- For higher j, pad the measure with additional true / false literals to ensure only one part can be played

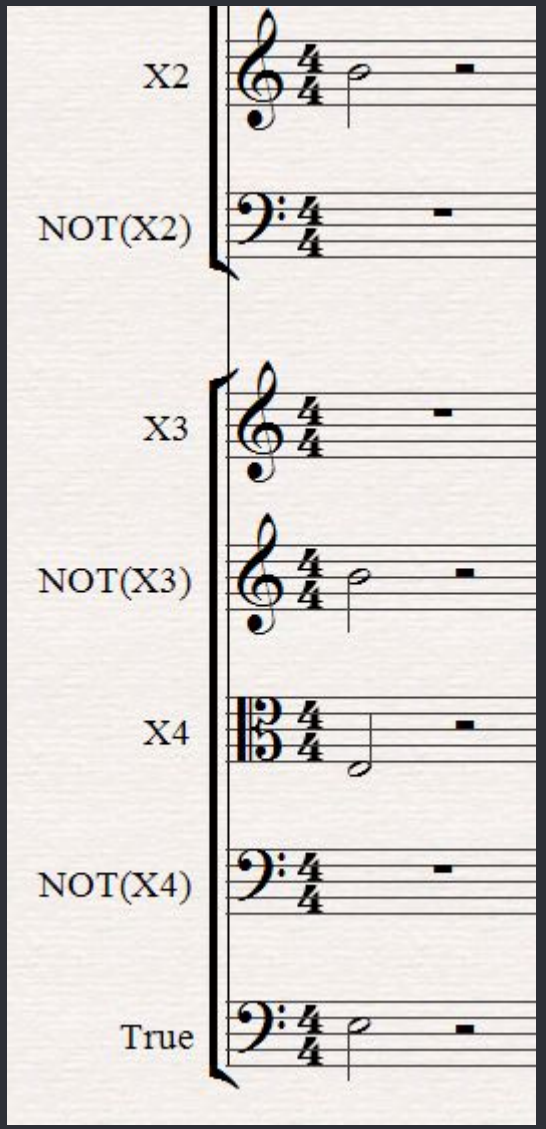


Summary of Results

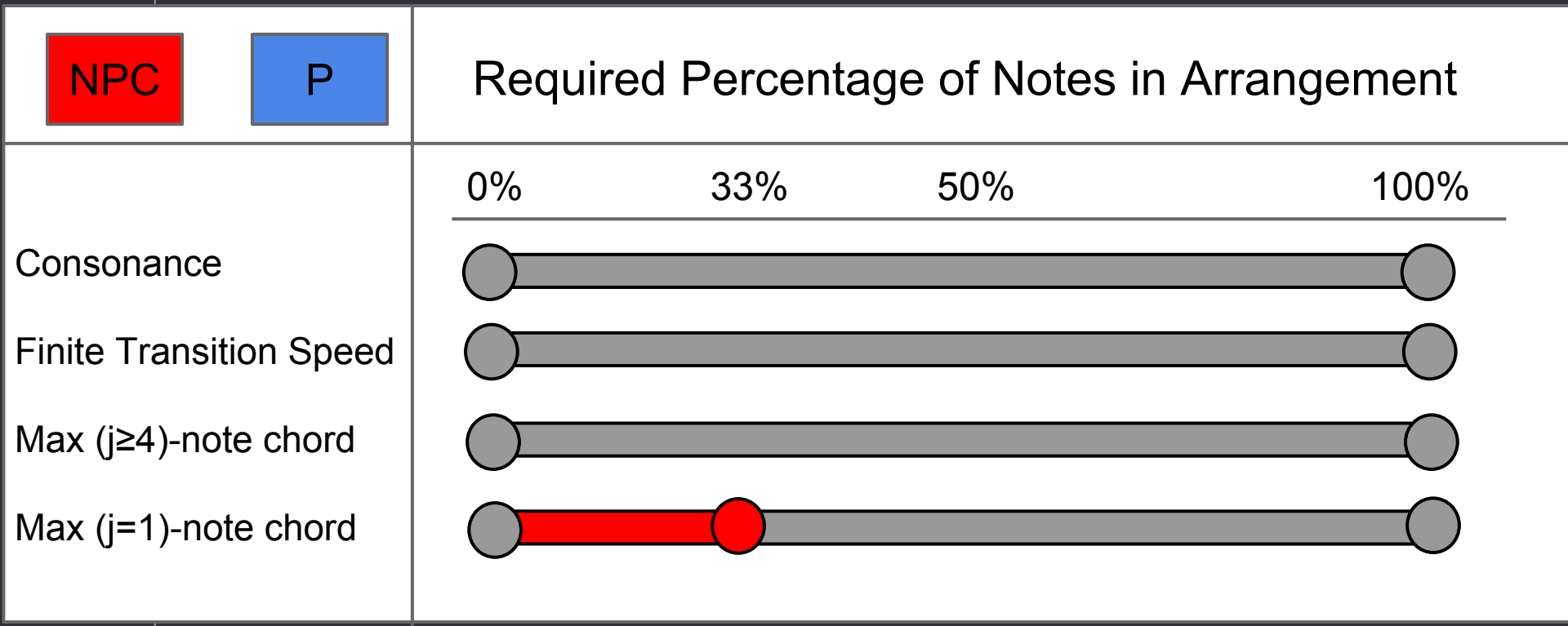


Clause Gadgets (j >= 4)

- Clause represented by measure with many variable / literal tracks playing simultaneously, but must choose subset of at most j
- Pad the measure with additional true / false literals to ensure only one variable can be played and still meets minimum number of note requirement

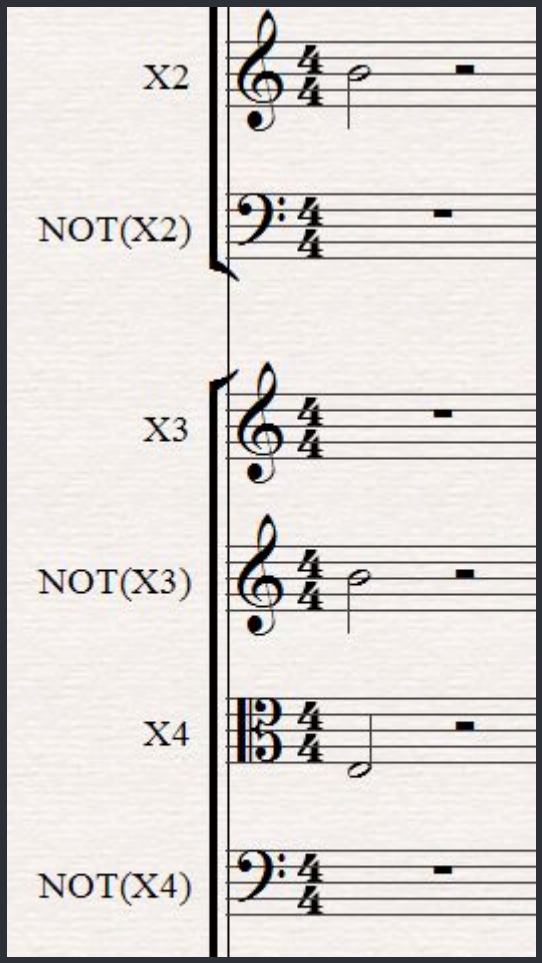


Summary of Results

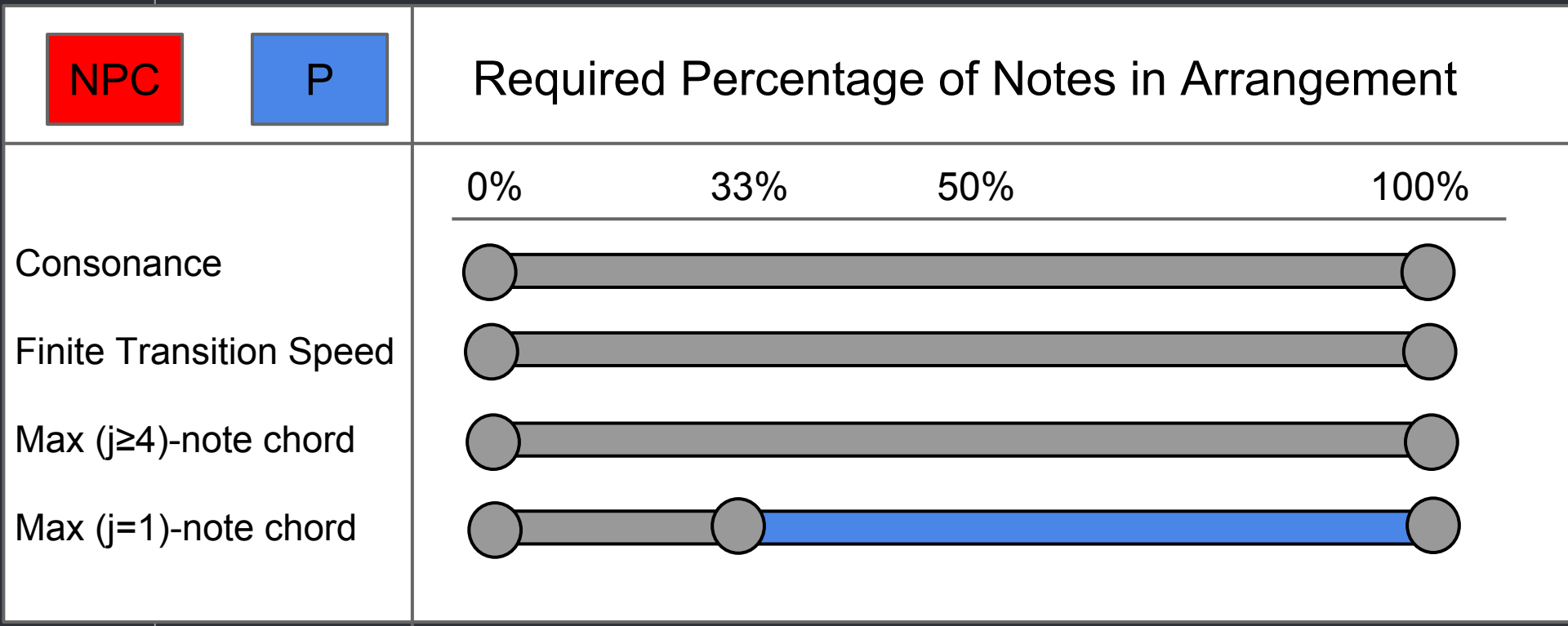


Clause Gadgets (j = 1), ⅓ notes

- For case where at least n% (0<n<=33.3) of notes must be played:
- Clause represented by measure with 3 variable tracks playing simultaneously, but can only play 1
- Must play 1 from original intent requirement
- Still NP-hard

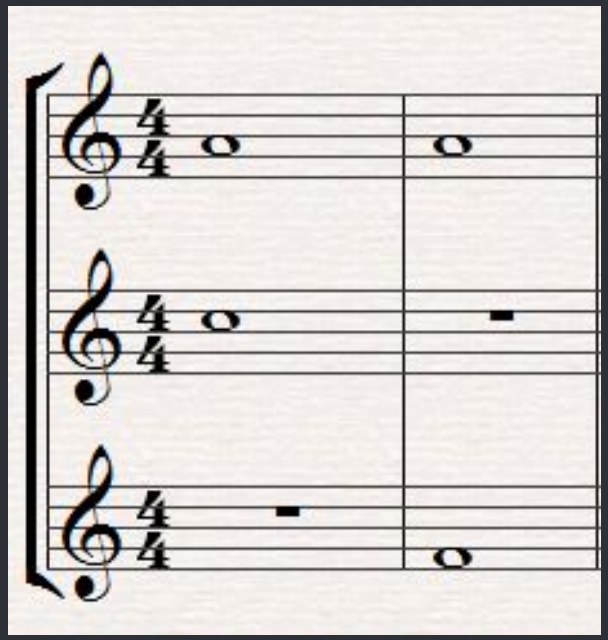


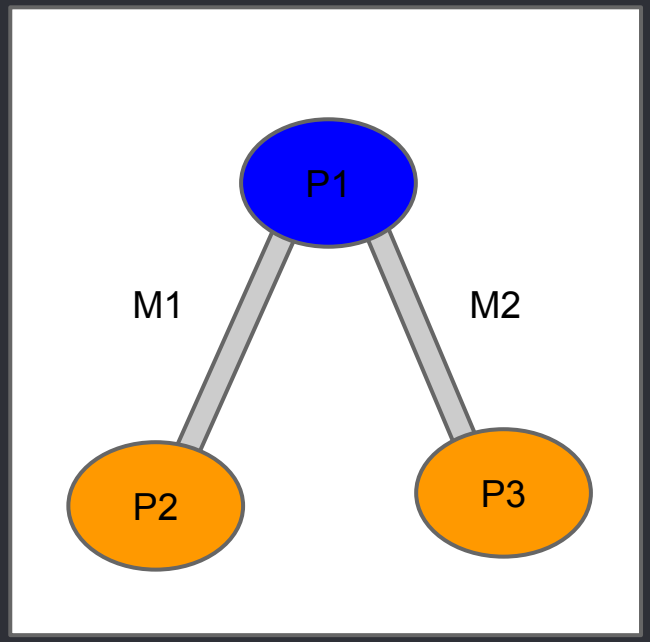
Summary of Results



j = 1, n > ⅓ notes (P)

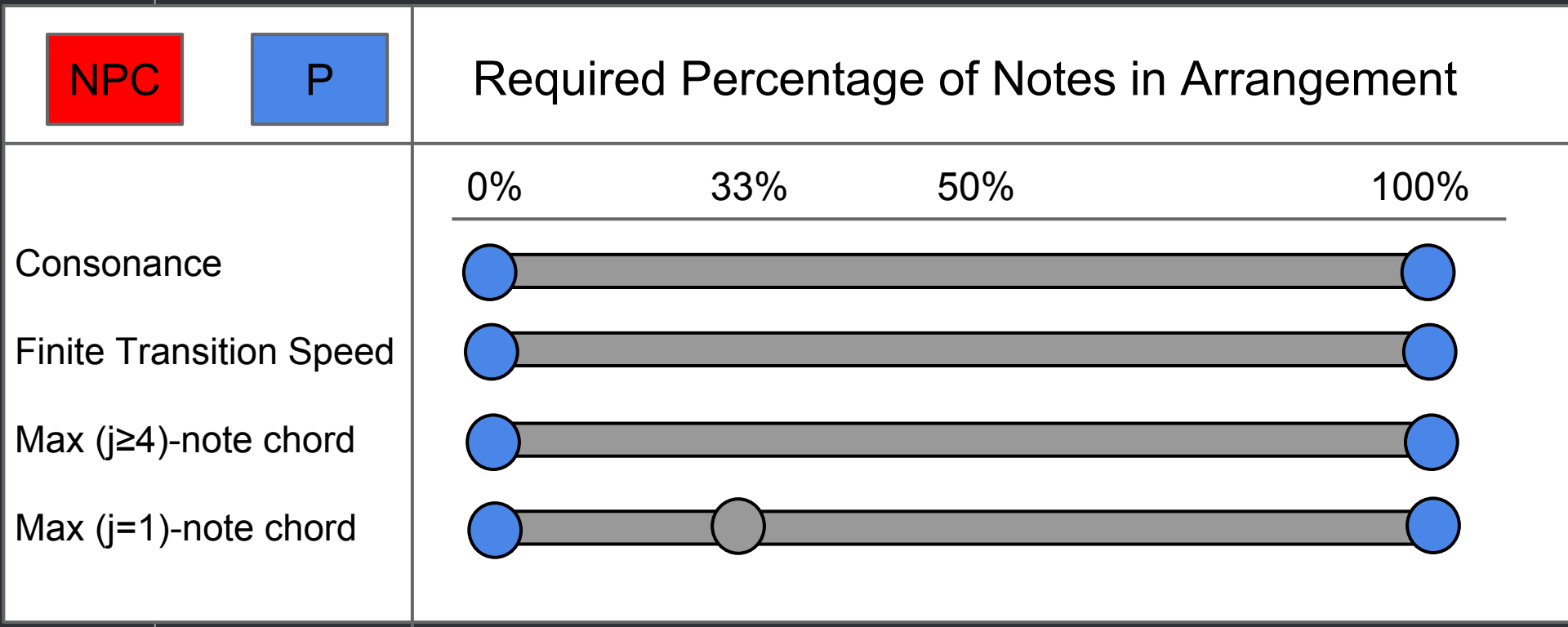
- Solvable in polynomial time via 2-coloring, with two colors: played and not played





Edge Cases

Summary of Results



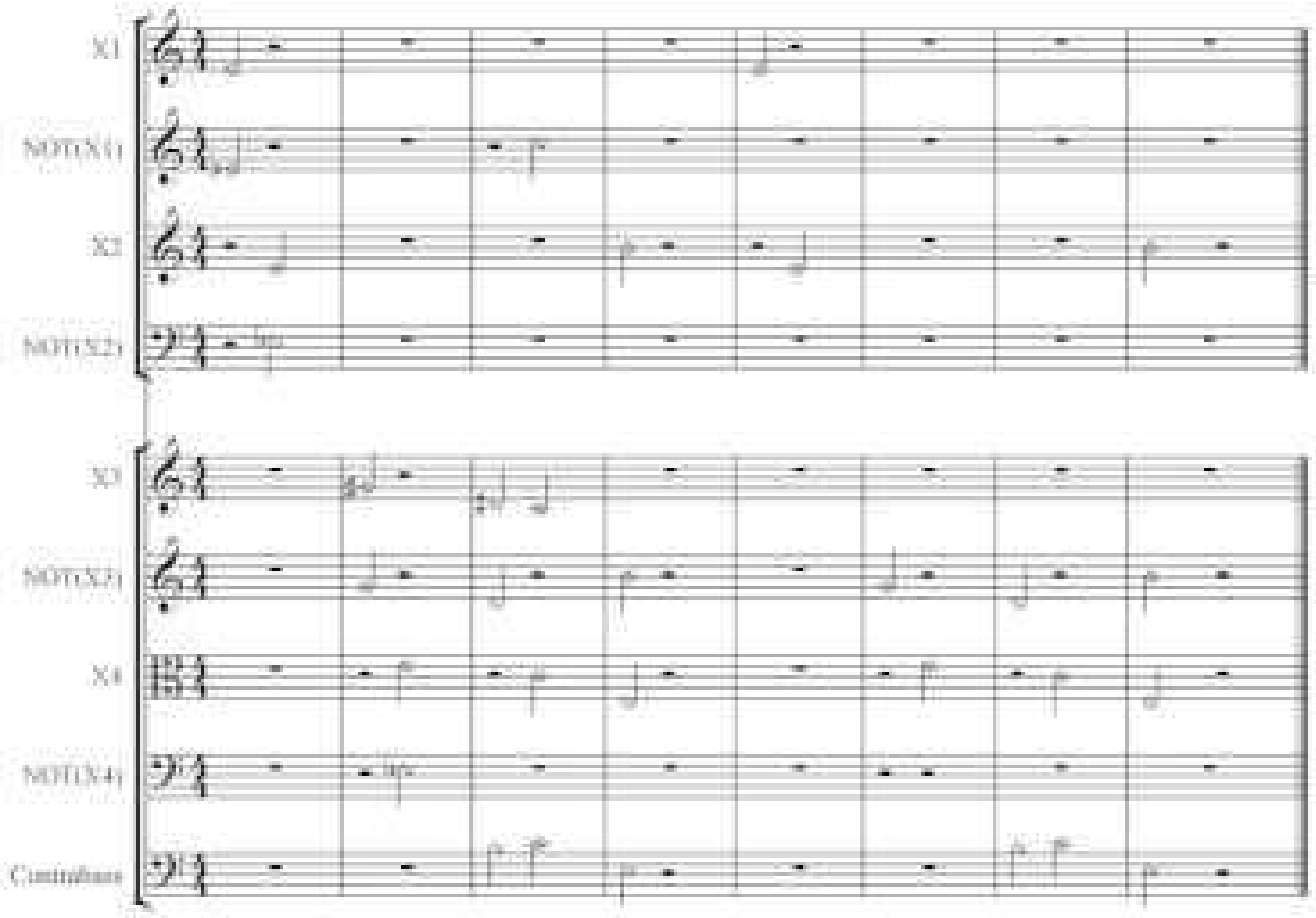
Edge Cases

- Allowing n=0 for the requirement that n% of notes to be played makes the problem P since you can simply play no notes
- Likewise allowing n=100 makes the problem P since you must select all tracks and thus can check that no other conditions are violated in a linear scan through the notes played

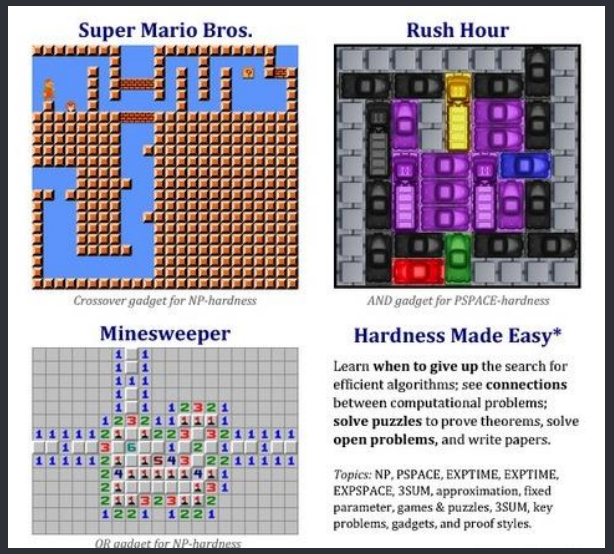
Fun Applications

Applications

- Creating rhythm game (Rock Band, Dance Dance Revolution) tracks is thus NP-hard
- Choreographing dance and fight scenes in music is thus NP-hard.


X1
NOT(X1)
X2
NOT(X2)
X3
NOT(X3)
X4
NOT(X4)
Contrabass

Questions?



NPC

P

Required Percentage of Notes in Arrangement