Fast and accurate resolution of ecDNA sequence using Cycle-Extractor

Mahsa Faizrahnemoon[1], Jens Luebeck[1], King L. Hung[2], Suhas Rao[1,3], Gino Prasad[1], Ivy Tsz-Lo Wong[6], Matthew G. Jones[4], Paul S. Mischel[5,6], Howard Y. Chang[7], Kaiyuan Zhu[1,8,*], and Vineet Bafna[1,9,*]

[1]Department of Computer Science & Engineering, UC San Diego, La Jolla, CA, USA
[2]Department of Neuroscience, Scripps Research, La Jolla, CA, USA
[3]Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA
[4]Department of Biology, Massachusetts Institute of Technology, Cambridge, MA, USA
[5]Sarafan ChEM-H, Stanford University, Stanford, CA, USA
[6]Department of Pathology, Stanford University School of Medicine, Stanford, CA, USA
[7]Amgen Research, South San Francisco, CA, USA
[8]School of Computer Science, Shanghai Jiao Tong University, Shanghai, China
[9]Halicioğlu Data Science Institute, UC San Diego, La Jolla, CA, USA
[*]Correspondence: k9zhu@ucsd.edu, vbafna@ucsd.edu

Abstract

Extrachromosomal DNA (ecDNA) plays a key role in cancer pathology. EcDNAs mediate high oncogene amplification and expression and worse patient outcomes. Accurately determining the structure of these circular molecules is essential for understanding their function, yet reconstructing ecDNA cycles from sequencing data remains challenging. We introduce Cycle-Extractor (CE) for reconstruction. CE accepts a breakpoint graph derived from either short or long read sequencing data as input and extracts a cycle with the maximum length-weighted-copy-number. CE utilizes a mixed-integer linear program (MILP) and a separate traversal procedure, enabling fast optimization and compatibility with free solvers.

We evaluated CE against CoRAL (long-read-based quadratic optimization), Decoil (long-reads), and AmpliconArchitect (AA for short reads) on both simulated data and real cancer cell lines. On simulated ecDNA, CE achieves performance comparable to CoRAL across three accuracy metrics and consistently outperforms AA and Decoil. On cancer cell lines, CE produces longer and heavier cycles than AA, and achieves performance similar to CoRAL. Moreover, CE is, on average, 40× faster than CoRAL. These results demonstrate that CE accurately reconstructs ecDNA from both short- and long-read sequencing data, while long-read inputs allow CE to recover more complete and higher-confidence ecDNA structures. CE improved the prediction of many ecDNA structures. On a PC3 ecDNA containing *MYC*, CE uses ONT data to reconstruct a substantially larger and higher-copy sequence (4.2 Mbp) compared to the short-read-derived reconstruction (690 Kbp). CRISPR-CATCH experiments confirm the presence of a large ecDNA molecule, validating the long-read-based CE reconstruction.

Keywords: extrachromosomal DNA (ecDNA); cancer genomics; breakpoint graphs; cycle reconstruction; mixed-integer linear programming (MILP); long-read and short-read sequencing

1 Introduction

Extrachromosomal DNA (ecDNA) is the most common source of amplified oncogenes across human cancers[1–4]. Studies on large cohorts have revealed that ecDNAs are present in approximately 14%-17% of tumors across multiple cancer types, with particularly high prevalence in glioblastoma and other aggressive malignancies[4,5]. These circular DNA molecules, typically ranging from kilobases to megabases in size[4], drive tumor formation[6], evolution[1,7–9] and therapeutic resistance[10–12]. Unlike chromosomal amplifications, ecDNAs lack centromeres and segregate asymmetrically during cell division[13,14], leading to rapid changes in copy numbers and intratumoral heterogeneity[8]. The unique properties of ecDNAs, including their ability to amplify oncogenes[4,5], enhance chromatin accessibility[15], and facilitate long-range regulatory interactions[16], together make them powerful mediators of tumor adaptation and progression.

To harness the clinical potential of ecDNA, two distinct, albeit related capabilities are needed: *detection* and *reconstruction* of ecDNA sequences. Determining whether or not the tumor genome has ecDNA can provide prognostic information for treatment. Clinical trials are underway for drugs that specifically treat patients with ecDNA[17], and detection of ecDNA is required for patient selection. Moreover, large landscape pan-cancer studies correlate ecDNA presence with other phenotypic variables, providing insight into the phenotypic consequence of ecDNA during tumor progression[4,5]. Reconstruction of the ecDNA sequence, on the other hand, is necessary to understand the content of ecDNA. The ecDNA sequence information has been used to identify oncogenes amplified by ecDNA[5], immunomodulatory genes[4], ecDNA with only regulatory sequences[4], ecDNA-ecDNA interaction[18], enhancer hijacking[19], and the 3-dimensional structure of ecDNA[20]. We focus here on the reconstruction problem.

Whole genome sequencing (WGS) based approaches have become the standard method for both detection and reconstruction problems. The current computational methods typically follow a parallel strategy for both, starting first with construction of a *breakpoint (or amplicon) graph* that represents amplified genomic regions and their novel adjacencies, using mapped WGS reads[21,22]. For detection, algorithms identify ecDNA by searching for cyclic paths in the graph that exhibit high copy number amplification[5,23].

Because the graph encodes all breakpoints participating in the ecDNA, the reconstruction problem corresponds to extraction of (potentially non-simple) cycles from the amplicon graph. For reconstruction, these cycles must be accurately extracted from the graph to determine the complete ecDNA sequence and structure. Therefore, unlike diploid genome assembly, ecDNA reconstruction faces the following unique challenges. First, ecDNAs can exhibit high structural complexity, containing many breakpoints that create complex rearrangements[24,25]. Some breakpoints can be missed by current sequencing and mapping technologies, particularly when they overlap with low-complexity or centromeric regions[26–29]. Second, large genomic segments can be duplicated within a single ecDNA molecule, appearing multiple times, with potentially different orientations and contexts, which complicates the determination of the correct order of segments[18,20,27]. Third, ecDNA typically occurs as a heterogeneous population - multiple distinct ecDNA species often coexist within the same tumor[30]. When these species share genomic segments, accurate reconstruction and copy-number estimation become challenging.

Recent developments focus on reconstruction utilizing long-reads (ONT, Pacbio)[27,31]. Long-reads improve cycle extraction by (a) better mapping of all breakpoints, including those that span repetitive or low-complexity regions, and (b) occasionally spanning regions of high multiplicity reducing ambiguities in reconstruction. CoRAL[27] specifically focused on the problem of extracting cycles that best explain the total copy number and breakpoints in the amplicon graph (see Results), and showed reasonable results on simulations and on cancer cell-lines. However, it uses a mixed

integer quadratically constrained programming (MIQCP) that is computationally demanding, and lowers utility for users with no access to MIQCP solvers. Here, we develop Cycle-Extractor(CE), which transforms all quadratic constraints to Mixed Integer Linear Programming (MILP) formulations, improving running time by an order of magnitude while maintaining or exceeding its performance. We tested CE on extensive simulations and on cell-lines, and also tested its ability to detect heterogeneity, where multiple ecDNAs share genomic regions. Finally, by abstracting and separating cycle extraction from graph reconstruction, we could apply CE to better predict ecDNA sequences using only short-read data, increasing its utility for analysis of complex rearrangements.

2 Methods

2.1 A high-level description of Cycle-Extractor

Cycle-Extractor (CE) takes an *amplicon graph* (or *CN-weighted breakpoint graph*) and a set of *subwalk constraints* as input (Figure 1), both defined below. The (undirected) amplicon graph is described by $G = (V, E, C, \ell)$, where the edge set $E = E_s \cup E_d \cup E_c$ is the union of *sequence, concordant,* and *discordant* edges. Sequence edges correspond to unbroken chromosomal segments in ecDNA. Discordant edges connect the beginning/end of one sequence edge to itself (indicating *inverted duplication* or *foldback*), or another sequence edge located elsewhere in the genome; while concordant edges connect the end of one sequence edge to the beginning of an adjacent sequence edge on the same chromosome. Each edge (u, v) is associated with a copy-number C_{uv}, and length ℓ_{uv}. The length of the segment edge is the number of nucleotides in the corresponding segment, while $\ell(u, v) = 0$ for all $(u, v) \in E_c \cup E_d$.

CE only allows *walks* W in G in which segment edges alternate with discordant or concordant edges. A walk W is cyclic if it begins and ends at the same vertex. A walk W with copy-number C_W is *feasible* if the copy number of each edge (u, v) used in the walk satisfies $C_W \cdot m_{uv} \leq C_{uv}$, where m_{uv} is the number of times an edge (u, v) is used in W (i.e., its *multiplicity*). The *length-weighted-copy-number* of a walk is defined by $\text{LWCN}(W, C_W) = C_W \cdot \text{length}(W)$. CE optimizes over feasible cyclic walks to output the *heaviest* (maximizing $\text{LWCN}(W, C_W)$) cyclic walk and maximize the number of subwalk constraints satisfied. At a high level, CE has two steps. In the first, *optimization* step, it iteratively computes C_W and an Eulerian subset of edges (u, v), along with their multiplicities, that comprise the heaviest cyclic walk[1]. Then the copy numbers from these edges are removed from the graph, and the next iteration proceeds until most of the length weighted copy number the graph G ($\geq \alpha \cdot \text{LWCN}(G)$, by default we set $\alpha = 0.9$) is consumed. In the second, *traversal* step, it identifies the actual ordering of the edges in each walk.

One key advantage of long reads is that they provide subwalk constraints that guide reconstruction. Long reads sampled from a cyclic walk W are subwalks of W. However, real samples may exhibit heterogeneity of structure, where multiple ecDNA may have similar but non-identical sequences, and subwalks may be sampled from different cyclic walks. We model subwalks using $p^j_{uv} \in \{0, 1\}$ indicating if edge (u, v) is in subwalk P_j. We then incorporate constraints corresponding to each subwalk. Given a set of subwalk constraints, $\mathcal{P}$, CE finds a feasible alternating cycle that simultaneously maximizes LWCN and the number of satisfied subwalk constraints.

2.2 The optimization step of CE

The CE MILP utilizes input encodings and a small number of tunable parameters:

[1]While CE prioritizes cyclic walks to reconstruct ecDNA, it can also identify non-cyclic walks when no heavy cycles can be found (see Supplementary Methods S1.3 for details).

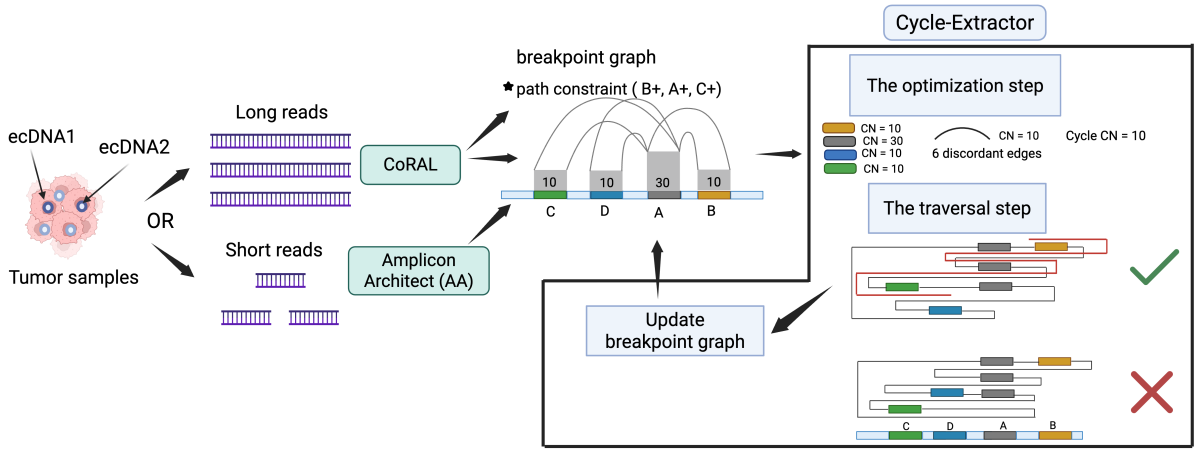


Figure 1: **Overview of Cycle-Extractor(CE).** CE takes as input, an amplicon graph generated using short-reads (e.g., from AA) or long-reads (e.g., from CoRAL). It applies a mixed-integer linear program to iteratively extract heaviest cycles from the graph (optimization step), and producing ordered traversals of edges (traversal step). From these traversals, CE selects the cycle that satisfies the maximum number of subwalk constraints. The graph is updated by decreasing the copy number of the edges in the selected cycle, and the process is repeated until the amplification is explained.

M**:** maximum multiplicity allowed for concordant and discordant edges

$\gamma(W)$**:** $= 0.01 \sum_{(u,v)\in W} C_{uv}\ell_{uv}$, and balances weights of subwalk constraints relative to LWCN in the objective.

$p^j_{uv} \in \{0,1\}$: indicates if edge (u,v) is in subwalk p_j

Variables. $F \in \mathbb{R}$: copy number assigned to edges in the solution but with multiplicity one.

$x_{uv} \in \{0,1\}$: $x_{uv} = 1$ if edge (u,v) is selected, and 0 otherwise.

$f_{uv} \in \mathbb{R}$: copy number assigned to edge (u,v) in the solution, should be multiples of F.

$y_{uvm} \in \{0,1\}$: (auxiliary) $y_{uvm} = 1$ iff (u,v) has multiplicity $m \in \{1 \le m \le M\}$ for each discordant edge in a feasible walk.

$z_{uvm} \in \mathbb{R}$: The auxiliary variable is used to constrain $z_{uvm} = m \cdot F \cdot y_{uvm}$ which is a quadratic formula but is linearized later.

$P_j \in \{0,1\}$: indicates if subwalk p_j is satisfied $1 \le j \le p$.

$d_u \in \{0,1,2,..\}$: indicates the order of the nodes, used in connectivity constraints.

$c^+_{uv} \in \{0,1\}$: be a binary variable that is 1 if the direction is from u to v, again used in connectivity constraints.

CE MILP. At each iteration, CE maximizes the sum of LWCN and γ-weighted subwalk constraint using

$$\max \sum_{(u,v)\in E} f_{uv}\ell_{uv} + \gamma \frac{\sum_j P_j}{|\mathcal{P}|}$$

while satisfying the following constraints.

1. **capacity**: For any edge (u,v) in a feasible solution, its copy number (f_{uv}) must not exceed its capacity (C_{uv}). If the edge is not chosen in the solution, its copy number is zero.

$$f_{uv} \le C_{uv} \cdot x_{uv} \quad \forall (u,v) \in E$$

2. **balance**: The sum of the solution copy number of discordant and concordant edges that connect to a node is equal to the solution copy number of the sequence edge connected to that node.

$$\sum_{(w,u)\in E_c\cup E_c} f_{wu} = \sum_{(u,v)\in E_s} f_{uv}$$

3. **copy number F with multiplicity one**: Note that if edge (u,v) is picked in the solution (i.e., $x_{uv}=1$), then $F \le f_{uv}$, but otherwise (u,v) imposes no constraint. For this purpose we introduce a sufficiently large constant $C_{\max}$ (e.g., $C_{\max} = M \cdot \max_{(u,v)\in E} f_{uv}$).

$$F \le f_{uv} + C_{\max} \cdot (1 - x_{uv}) \;\; \forall (u,v) \in E_c \cup E_d$$

4. **discordant edge multiplicity constraint**: The multiplicity of a discordant edge is managed using

$$\sum_{m=1}^{M} y_{uvm} = x_{uv} \qquad \forall (u,v) \in E_d.$$

Subsequently the copy of number of the edge is given by the (quadratic) constraint

$$f_{uv} = \sum_{m=1}^{M} m \cdot F \cdot y_{uvm} = \sum_{m=1}^{M} z_{uvm} \qquad \forall (u,v) \in E_d$$

which guarantees f_{uv} are multiples of F. Due to the balance condition, we can prove that f_{uv} is multiple of F for each concordant and each sequence edge (Supplementary Methods).

5. **subwalk constraints**: If a subwalk is in the solution ($P_j = 1$), all of its edges are in the solution. If a subwalk in not in the solution ($P_j \neq 0$), there is not constraint here.

$$x_{uv} \ge P_j \cdot p^j_{uv} \qquad \forall j \in \{1, ..., p\} \qquad \forall (u,v) \in E \cap p_j$$

where $p^j_{uv} \in \{0,1\}$ is a constant indicating if edge (u,v) is in subwalk p_j.

Additional constraints including the linearization of discordant edge multiplicity constraint can also be found in Supplementary Methods.

2.3 Resolving multiple disconnected cycles

CE maximizes the length-weighted-copy-number, aiming to identify the heaviest solution. With no connectivity requirements, it can generate multiple disconnected cycles within a single iteration, as long as LWCN improves. However, this behavior can result in a reduction of the copy numbers for some cycles. For example, if there are two disjoint cycles with length 100 Kbp each, and CN 20 and 40, respectively, CE could report a 'single' solution outputting both cycles with CN 20 (LWCN=4M). To prevent this, we implemented two methods.

In the first, *connectivity-enforcement* method, we added MILP constraints to enforce that exactly one cycle with maximum LWCN is output per iteration (Supplemental Methods). In the previous example, the constraints ensure that only one cycle with CN=40 is output. The second cycle and other cycles are output one at a time in future iterations. In the second, *CN-enhancement* method, we allowed the MILP to output multiple disjoint cycles initially, but subsequently increase the CN of each cycle to the minimum copy number of any edge in that cycle. Among all cycles initially produced, we output the cycle with the maximum enhanced copy number, and reiterate.

In the Results section below, we refer to connectivity-enforcement method as CEc, and the CN-enhancement method as CE, treating it as the default. We tested CE and CEc for accuracy and speed.

2.4 The traversal step of CE

Given a cycle returned in the optimization step as a collection of edges $\mathcal{W}$ and their multiplicities $m_{uv} = m = \frac{f_{uv}}{F}$ where $y_{uvm} = 1$, we aim to construct an Eulerian traversal as an ordered and signed list of edges, such that each edge $(u, v) \in \mathcal{W}$ occurs m_{uv} times, and the traversal satisfies the maximum number of subwalk constraints. Since exhaustive enumeration of all Eulerian traversals can be intractable[32], CE samples from the space of all Eulerian traversals using a modified Hierholzer's algorithm, and picks one traversal that satisfies the most subwalk constraints.

Specifically, CE maintains a count r_{uv} of the number of times each edge (u, v) remains unused, initialized as its multiplicity m_{uv}. It starts with the sequence edge $(u, v) \in E_s$ with positive r_{uv} that begins at the canonically smallest chromosome and position, with positive orientation (i.e., starting with the node corresponding to the smaller position). The algorithm then alternates between selecting a non-sequence edge (discordant or concordant) and a sequence edge. When a non-sequence edge is to be chosen, CE randomly selects a discordant or concordant edge (adjacent to the current node) with $r_{uv} > 0$ and heads to the next node. Since each node connects to only one sequence edge, the next sequence edge is automatically determined; and if the node corresponds to the larger endpoint in the sequence edge, the orientation is negative, otherwise it is positive. This process continues until the walk returns to its starting node, thereby forming a closed walk. Finally, all closed walks are iteratively merged into a single walk by identifying shared nodes and rotating the edges for the cycle to be inserted.

3 Results

Datasets and metrics. We simulated 75 distinct cyclic structures using ecSimulator (`https://github.com/AmpliconSuite/ecSimulator`) closely following CoRAL[27]. Very briefly, each structure contains a collection of between 1 and 20 oriented genomic segments joined end to end. The end of the last segment is joined to the beginning of the first segment to complete the cyclic structure. The simulated structures modeled the known mechanisms of ecDNA formation including episomal[7], chromothripsis[25,33], and 2-foldback[34,35], and were additionally modified using internal structural variations. Next, long and short reads were simulated from the simulated ecDNA sequence at one of three coverages (50X, 100X, or 250X) modeling the increased copy number of ecDNA, and merged with reads from one of five simulated normal, diploid genomes (each with 13X coverage).

From long-read input, an amplicon graph was generated using CoRAL. Cycle reconstruction was performed using CoRAL and Cycle-Extractor for comparisons. For short-reads, the amplicon graph was generated using AA, and cycle extraction was performed using AA and Cycle-Extractor.

We additionally analyzed 31 amplicon graphs from cancer cell-lines (Supp. Table S2), 26 with Illumina WGS, and 12 amplicons with long-reads (ONT) whole genome sequencing, with both sequencing modalities for 7 amplicons. All cell-lines are known to carry ecDNA from prior cytogenetics experiments. For the cell-lines, the true ecDNA sequence is not known, and therefore, we tested them on the previously defined LWCN (length-weighted-copy-number) and LWCNR (ratio) metrics. The tools are all designed for the user to have some control on the stopping criteria. For all experiments below, and all methods (except AA), we stopped when 90% of the total LWCN in the graph was explained, or if the current iteration had not solved the MILP after 2 hours. For AA, such a control is not available, and the default mode was used.

CE matches or exceeds CoRAL, Decoil and AmpliconArchitect (AA) performance in simulations. No single metric captures the quality of reconstructed cycles, and we utilized 3 metrics: First, to measure the nucleotide overlap, consider the true and reconstructed cycles each

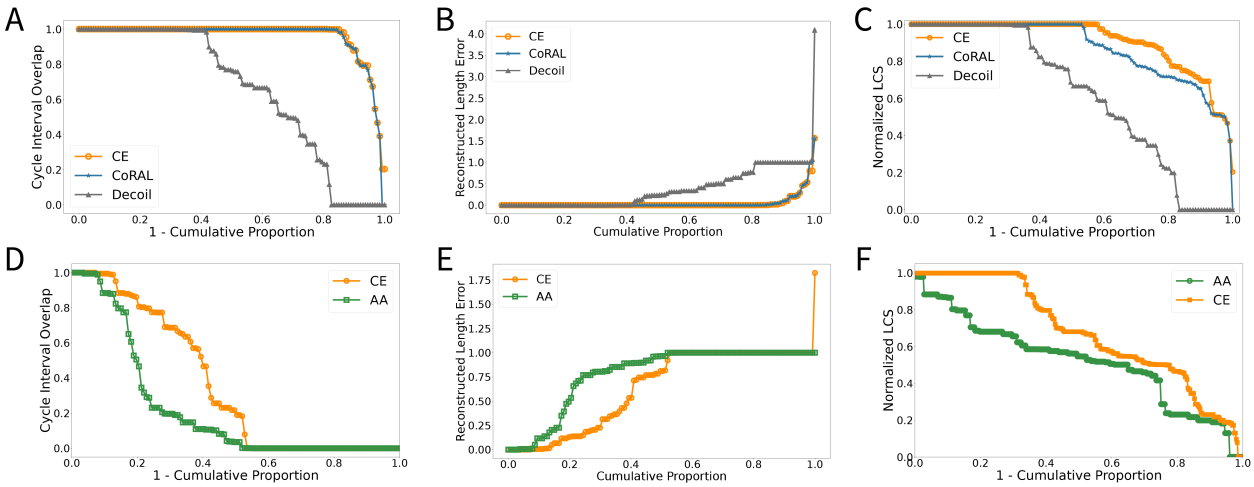


Figure 2: **CE accuracy in reconstruction on simulated data.** (A-C): CIO, RLE, and LCS metrics comparing CE to CoRAL and Decoil on long-read simulations. For any threshold of a metric the results show the fraction that matched or improved upon the metric. (D-F): Comparison between CE and AA on short-read simulations.

as multi-sets T, R (including multiplicities) of nucleotide positions. The **Cycle Interval Overlap (CIO)** metric given by

$$\mathrm{CIO}(T, R) = \frac{|T \cap R|}{|T \cup R|},$$

represents the Jaccard index of the shared nucleotides. Next, **Reconstruction Length Error (RLE)** reports the normalized difference in amplicon lengths computed as

$$\left(\frac{|L_t - L_r|}{|L_t|}\right),$$

where L_t and L_r denote the lengths of the true and reconstructed cycles, respectively. Finally, **Cyclic Longest Common Subsequence (LCS)** measures the length of the longest common subsequence contained in the true cycle and the reconstructed cycle after eliminating intervals that are not found in both, and normalizing to the length of the true cycle.

For all three metrics, CE performed at least as well as CoRAL (Figure 2A–C) and outperformed Decoil[31] across all three metrics. As shown in Figure 2A–C, CoRAL also generally performed better than Decoil. Intriguingly, CE improved upon CoRAL for the LCS metric in nearly half of the cases. A partial explanation could be better satisfaction of subwalk constraints, where CE satisfied 7.6% more subwalk constraints than CoRAL. Additionally, CoRAL makes a design choice which gives the same priority to extracting walks and cycles. In some cases where a breakpoint edge is missed, CoRAL selects a walk that is heavier than any available cycle. In summary, the CoRAL and CE reconstructions, while not identical, match up well overall.

For short-reads, CE performed much better than AA on all metrics, showcasing the power of optimization in extracting the heaviest cycle (Figure 2D-F). While the amplicons used to simulate short and long-read data were the same, the long-read data missed fewer breakpoints and also provided subwalk constraints. Therefore, the cumulative performance of CE on short-reads was worse than its own performance on long-reads across all three metrics (Supplementary Figure S1). Manual exploration confirmed that this was due to missed breakpoints in short-read analysis, showcasing the additional sensitivity of long-reads for breakpoint detection. In some simulations,

missed breakpoint edges prevented cycle formation (CIO=0 and RLE=1) or shortened the cycle. This happened often with short-reads but sometimes also for long-reads. The cyclic LCS metric could be dominated by score of a long, unbroken segment even when the breakpoint ordering elsewhere was incorrect. To assess if this inflated our LCS metrics, we plotted the distribution of segment lengths and ecDNA sizes (Supplementary Figure S2). A large majority of the fragments were < 250Kbp, much smaller than the average size of ecDNA (2Mbp), discounting long fragments as impacting the score. Moreover CE had 100% accuracy in 62% of the cases, possible only with a perfect reconstruction.

Finally, we also tested CE on additional difficult cases where two true cycles sharing genomic segments were simulated. When the two cycles had very similar copy numbers, CE merged them into a longer, heavier cycle (Supplementary Figure S3A). However, when the copy numbers were very different, CE predicted the two cycles accurately (Supplementary Figure S3B). Heterogeneity of ecDNA is not well studied and CE provides a method for its exploration in real data.

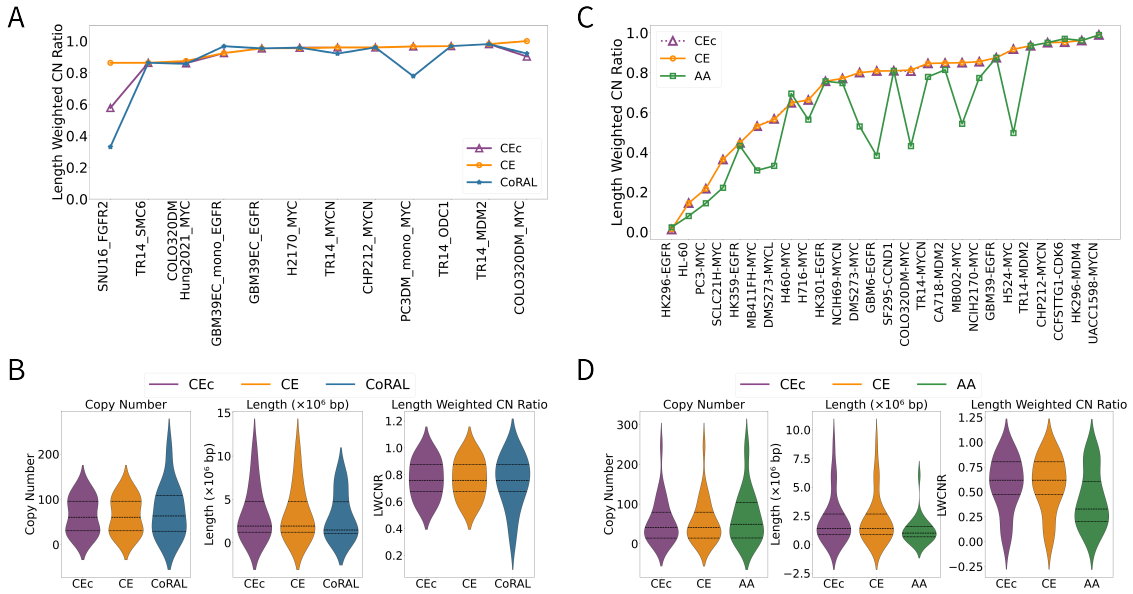


Figure 3: **CE performance on cancer cell-lines.** A. Comparison of LWCNR of the three heaviest cycles extracted by CE, CEc and CoRAL on ONT sequenced cell-lines. B. Copy Number, length and LWCNR of the heaviest cycle on ONT derived graphs. C. LWCNR of the 3 heaviest cycles extracted by CE, CEc and AA on Illumina sequenced cell-lines. D. Copy Number, length and LWCNR of the single heaviest cycle on Illumina derived graphs.

CE reconstructs heavier cycles in cancer cell-lines. To test performance on cell-lines where ecDNA have been previously reported, we used utilized whole genome ONT reads and used CoRAL to generate an amplicon graph for 12 ecDNA amplicons. We compared the cycle extractions of CE, CE with connectivity-enforcement (CEc), and CoRAL on the amplicon graphs. Because the cycle reconstructions may differ and produce different orderings, we chose the sum of the length weighted copy number ratio (LWCNR) for the three heaviest cycles identified by each method. The sum LWCNR describes how much of the copy number of the amplified region is explained by the three heaviest cycles.

CE, CEc and CoRAL exhibited nearly identical behavior (Figure 3A), both selecting cycles with high LWCNR as their top-ranked cycles, and generally explaining a large fraction of the amplicon.

SNU16 was the single exception where both CEc and CoRAL extracted only one cycle when the execution was stopped, while CE extracted 5 cycles and thus explained a larger proportion of LWCN. Notably, the one (heaviest) cycle extracted by CEc was identical to the heaviest cycle extracted by CE. In some other samples, CoRAL obtained lower LWCNR because it prioritized a heavy walk rather than a cyclic structure as the dominant amplicon. When considering all metrics, the total copy number, length, and LWCNR, comparing the three heaviest cycles is not meaningful, and we tested those for the single heaviest cycle. Here also, CE, CEc and CoRAL cycles matched up well (Figure 3B).

In contrast, when taking a short read amplicon graph as input, AA generally produced cycles with lower LWCNR (mean 0.598) compared to CE (mean 0.704) (Figure 3C,D). The performance of CE and CEc was effectively identical across all samples.

We also compared the performance of all methods using cell-lines for which both ONT and Illumina sequencing was available. Note that the copy number estimations, while comparable, are not identical between short and long-reads in these biological replicates, and LWCNR is not comparable. Therefore, we manually looked at LWCN, length, and CN of cycles produced on individual cell-lines (Supp. Figures S4-S8). In some cases (e.g., the CHP-212 ecDNA amplifying *MYCN*), the graph has a relatively simple structure. For simple cases, the cycles extracted by CoRAL and CE using the ONT graph and by AA and CE using the Illumina graph were all identical. In other cases, the cycles extracted by CoRAL and CE using the ONT amplicon graph were heavier and longer than the cycles extracted by AA and CE using Illumina. Moreover, even on the Illumina amplicon graph, the CE extracted cycle was heavier and longer than the AA extracted cycle. Together, these results indicate the power of using long-reads and a principled cycle extraction algorithm.

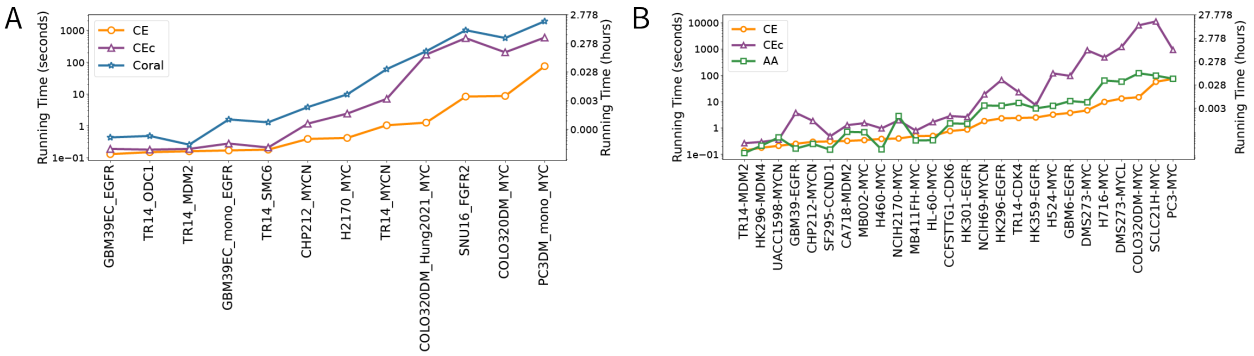


Figure 4: **Running times of different methods for cycle extraction.** A. Cycle extraction on ONT samples. B. Cycle extraction on Illumina samples. All cases, with the exception of SNU16 on ONT reads, terminated without reaching the time limit of 2 hours per iteration. The CoRAL and CEc running time for SNU16 are for extracting a single heaviest cycle.

CE runs 40× **faster than CoRAL.** CE was dramatically faster than other methods, completing in 0.13 to 75.47 seconds, with 58% of the samples finishing within one second (Figure 4A). In contrast, CoRAL required between 0.26 seconds to 32.5 minutes, and its mean running-time was 40× higher than CE. The running time of CEc was between 0.18 second to 10 minutes. CEc finished running 42% of samples within 1 second. Notably, all cases, with one exception, terminated without reaching the time limit of 2 hours per iteration. SNU16 was one challenging example with 70 discordant edges and 333 edges in total, where CoRAL and CEc extracted a single heaviest cycle before termination, and we reported the time for extracting the first cycle. However, CE only took

8.37 seconds to extract 5 cycles which explained 90.73% LWCN of the graph.

The results on Illumina short-reads were similar. CE continued to be the fastest method, running faster than AA on 75% of the samples despite the MILP based cycle extraction procedure (Figure 4B), while maintaining much higher reconstruction quality (heaviness). For CEc, application of connectivity enforcement method introduces additional (though, increasing linearly in the graph size) integer variables, increasing CEc running time compared to CE.

The running time of CEc is higher when using the short-read graph compared to its time on the long-read graph. Because of missing breakpoints, the cycles extracted are shorter, and more cycles are extracted, leading to higher running time for CEc using the short-read graph compared to the long-read graph.

The running time of CE increased with number of edges for long read simulated data (Supp. Figure S9A) and for data from cell lines (Supp. Figure S9B). While the running time depended upon graph complexity (number of edges), it was not the sole criterion. CE ran on complex amplicons like PC3 with 310 (133 sequence, 118 concordant, 59 discordant) edges in 75 seconds. Although the SNU16 graph contained more edges (332) than the PC3 graph, its running time was substantially shorter (under 10 seconds versus 75 seconds for PC3). The reason was that CE identified multiple disconnected cycles in SNU16 during early iterations, allowing the algorithm to satisfy the termination criterion (coverage of 90% of the graph) more quickly.

CE reconstructions improve our understanding of ecDNA biology. The algorithmic approach utilized by CE improves upon our ability to reconstruct ecDNA sequences. We compared some of the cycles predicted by AA and CE. In CA718, AA predicted 3 cycles, one carrying *MYCN* and *MDM2*, another carrying *PDGFRA*, and a third, non-genic cycle (Figure 5B). In contrast, CE generated had 2 cycles, effectively merging the non-genic cycle of AA with the *MDM2*+*MYCN* cycle (Figure 5C) to generate a longer and heavier cycle. The alternative explanation is meaningful and suggests that predictions of non-genic ecDNA containing only regulatory sequences[4] should be orthogonally validated.

We also compared the CE cycles generated on a PC3 amplicon using Nanopore and Illumina sequencing. The cycle reconstructed from Nanopore was substantially larger and more amplified than those obtained from the short-read data, predicting a 4.32 Mbp cycle (Figure 5C) explaining 78% of the amplicon copy number. In contrast, the cycle extracted using Illumina reads (Figure 5D) had length 690 Kbp and explained 9% of the amplicon. The CN of the oncogene *MYC* was 12 copies in the short-read cycle versus 47 copies in the long-read cycle. To validate our data, we performed a CRISPR-Catch experiment[30] where single CRISPR mediated cuts linearized the ecDNA, and pulse-field gel electrophoresis was used to separate the linearized product. Multiple guide RNAs were used to linearize the structure, and produced fragments with different lengths (Figure 5E). Notably, gRNA g3 produced fragments of lengths > 2.2, and 1.7 Mbp. On the predicted PC3 ecDNA, g3 occurred thrice, and supported fragments of size 2.5, 1.7, and 0.13 Mbp, consistent with the experimental data. The gRNA g5 is contained in the oncogene *MYC*. It appeared twice on the cycle and supported fragments of size 0.12 and 4.3 Mbp. The other gRNA were controls and did not appear on the cycle.

4 Discussion

Reconstruction of the ecDNA sequence is important to understand the content of ecDNAs identified in cancer genomes. Nevertheless, it remains a very challenging problem for two reasons. First, ecDNA can carry multiple larger regions with higher multiplicity, which is akin to the problem of

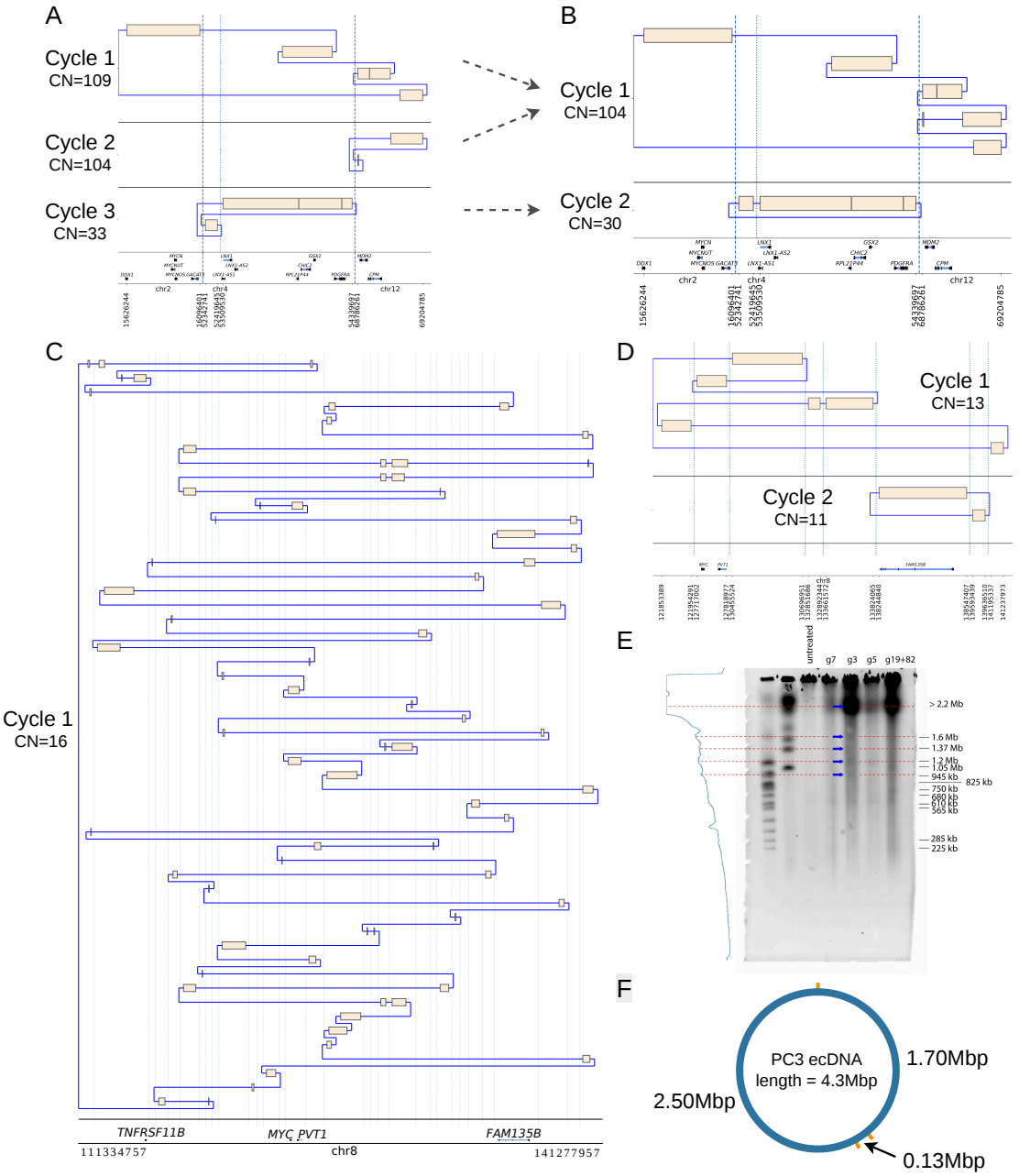


Figure 5: **AA and CE reconstructions on specific cell-lines.** A. AA reconstruction with 3 cycles for CA718. B. CE reconstrucion with 2 cycles. C. CE reconstruction for PC3 amplicon. D. CE reconstruction from AA (Illumina) graph. E. CRISPR Catch data for CE amplicon.

repeats in genome assembly, but harder because the regions of high multiplicity are very large often exceed 100 Kbp.

We describe an algorithmic approach wherein a cycle is extracted from the amplicon graph that best explains the copy number and breakpoints represents the current structure. Unlike previous methods, the CE methods utilizes an MILP optimization that is an order of magnitude faster while maintaining accuracy. Moreover, CE is designed to work off both Illumina and ONT reads and

can be easily adapted to other sequencing technology. On long-reads, it additionally optimizes the use of subwalk-constraints for better disambiguation of sequences. For both short and long read, the input amplicon graphs required as input to CE are now readily produced from wgs data using robust available tools[23,27], making it seamless to use.

In extensive tests, CE reconstructed ecDNA structures with high accuracy. Some metrics such as LWCNR could favor CE, which optimizes length weighted copy number (LWCN). However, when the ground truth (the true cycle) was known, for example in simulated data, we only used metrics that were independent of CE's objective function, thus providing a fair comparison for all methods.

When ground truth was not known, for example in cell-lines, LWCNR was the only evaluation metric that did not require access to ground truth data. We argued that it is a reasonable metric, because higher values suggest that a larger fraction of the amplified region was explained by one ecDNA, and that was more likely to correspond to biologically relevant structures. For many well studied lines, we obtained the identical answer to the previous (often manual) reconstructions from earlier studies. In one case where we did get a novel reconstruction, we performed a CRISPR Catch experiment for validation.

In ongoing future work, we will create modules for AmpliconSuite and CoRAL so that cycle extraction can be automatically performed using CE. By determining the ecDNA sequence, CE will have direct utility for investigating functional aspects of ecDNA including enhancer hijacking[19], regulatory rewiring, presence and absence of genes[4,5] and other functional elements, the DNA repair mechanisms used[36], evolution of ecDNA structure[6] and others.

The second challenge with ecDNA reconstruction is the heterogeneity where multiple ecDNA share genomic segments, and it is not clear if they represent the same ecDNA or different ones. Notably, those ecDNA could exist and function similarly as a single molecule or multiple molecules, or even both in one sample. Thus, even when heterogeneity exists, it is not easy to formulate as a problem of cycle extraction. In preliminary results, CE can extract two ecDNA sequences with shared segments if they have distinct copy numbers. We plan to extend this by developing the problem not as one of reconstruction but possibly about estimating the level of heterogeneity in terms of conflicting subwalk constraints.

In general, unambiguous determination of the correct ecDNA sequence will be best achieved by a combination of sequencing and other experimental methods. These include single molecule sequencing extensions such as CRISPR-Catch[30], single-cell methods like ATAC-seq[37] and Hi-C[20,38] and orthogonal technologies such as DNA-FISH[23] and others. These methods can be applied to individual samples of interest, and future work will seek to combine these methods with the algorithmic analysis of amplicon graphs. In the meantime, CE can serve as an easy to use and relatively accurate method for ecDNA reconstruction.

5 Code Availability

`https://github.com/AmpliconSuite/CycleExtractor`

References

[1] Kristen M Turner, Viraj Deshpande, Doruk Beyter, Tomoyuki Koga, Jessica Rusert, Catherine Lee, Bin Li, Karen Arden, Bing Ren, David A Nathanson, et al. Extrachromosomal oncogene amplification drives tumour evolution and genetic heterogeneity. *Nature*, 543(7643):122–125, 2017.

[2] Roel GW Verhaak, Vineet Bafna, and Paul S Mischel. Extrachromosomal oncogene amplification in tumour pathogenesis and evolution. *Nature Reviews Cancer*, 19(5):283–288, 2019.

[3] Xiaowei Yan, Paul Mischel, and Howard Chang. Extrachromosomal dna in cancer. *Nature Reviews Cancer*, 24(4):261–273, 2024.

[4] Chris Bailey, Oriol Pich, Kerstin Thol, Thomas BK Watkins, Jens Luebeck, Andrew Rowan, Georgia Stavrou, Natasha E Weiser, Bhargavi Dameracharla, Robert Bentham, et al. Origins and impact of extrachromosomal dna. *Nature*, 635(8037):193–200, 2024.

[5] Hoon Kim, Nam-Phuong Nguyen, Kristen Turner, Sihan Wu, Amit D Gujar, Jens Luebeck, Jihe Liu, Viraj Deshpande, Utkrisht Rajkumar, Sandeep Namburi, et al. Extrachromosomal DNA is associated with oncogene amplification and poor outcome across multiple cancers. *Nature genetics*, 52(9):891–897, 2020.

[6] Jens Luebeck, Alvin Wei Tian Ng, Patricia C Galipeau, Xiaohong Li, Carissa A Sanchez, Annalise C Katz-Summercorn, Hoon Kim, Sriganesh Jammula, Yudou He, Scott M Lippman, et al. Extrachromosomal DNA in the cancerous transformation of Barrett's oesophagus. *Nature*, pages 1–8, 2023.

[7] Susan M Carroll, Margaret L DeRose, Patrick Gaudray, Charleen M Moore, Donald R Needham-Vandevanter, Daniel D Von Hoff, and Geoffrey M Wahl. Double minute chromosomes can be produced from precursors derived from a chromosomal deletion. *Molecular and cellular biology*, 8(4):1525–1533, 1988.

[8] Joshua T Lange, John C Rose, Celine Y Chen, Yuriy Pichugin, Liangqi Xie, Jun Tang, King L Hung, Kathryn E Yost, Quanming Shi, Marcella L Erb, et al. The evolutionary dynamics of extrachromosomal DNA in human cancers. *Nature genetics*, 54(10):1527–1533, 2022.

[9] Imran Noorani, Magnus Haughey, Jens Luebeck, Andrew Rowan, Eva Grönroos, Francesco Terenzi, Ivy Tsz-Lo Wong, Davide Pradella, Marta Lisi, Jeanette Kittel, et al. Extrachromosomal dna–driven oncogene spatial heterogeneity and evolution in glioblastoma. *Cancer Discovery*, 15(10):2078–2095, 2025.

[10] Daniel A Haber, Stephen M Beverley, ML Kiely, and Robert T Schimke. Properties of an altered dihydrofolate reductase encoded by amplified genes in cultured mouse fibroblasts. *Journal of Biological Chemistry*, 256(18):9501–9510, 1981.

[11] David A Nathanson, Beatrice Gini, Jack Mottahedeh, Koppany Visnyei, Tomoyuki Koga, German Gomez, Ascia Eskin, Kiwook Hwang, Jun Wang, Kenta Masui, et al. Targeted therapy resistance mediated by dynamic regulation of extrachromosomal mutant egfr dna. *Science*, 343 (6166):72–76, 2014.

[12] Shreoshi Pal Choudhuri, Luc Girard, Jun Yi Stanley Lim, Jillian F Wise, Braeden Freitas, Di Yang, Edmond Wong, Seth Hamilton, Victor D Chien, Yoon Jung Kim, et al. Acquired

cross-resistance in small cell lung cancer due to extrachromosomal dna amplification of myc paralogs. *Cancer discovery*, 14(5):804–827, 2024.

[13] King L Hung, Matthew G Jones, Ivy Tsz-Lo Wong, Ellis J Curtis, Joshua T Lange, Britney Jiayu He, Jens Luebeck, Rachel Schmargon, Elisa Scanu, Lotte Brückner, et al. Coordinated inheritance of extrachromosomal dnas in cancer cells. *Nature*, 635(8037):201–209, 2024.

[14] Venkat Sankar, King L Hung, Aditi Gnanasekar, Ivy Tsz-Lo Wong, Quanming Shi, Katerina Kraft, Matthew G Jones, Britney Jiayu He, Xiaowei Yan, Julia A Belk, et al. Genetic elements promote retention of extrachromosomal dna in cancer cells. *bioRxiv*, pages 2025–10, 2025.

[15] Sihan Wu, Kristen M Turner, Nam Nguyen, Ramya Raviram, Marcella Erb, Jennifer Santini, Jens Luebeck, Utkrisht Rajkumar, Yarui Diao, Bin Li, et al. Circular ecDNA promotes accessible chromatin and high oncogene expression. *Nature*, 575(7784):699–703, 2019.

[16] Kathryn E Yost, Yanding Zhao, King L Hung, Kaiyuan Zhu, Duo Xu, M Ryan Corces, Shadi Shams, Bryan H Louie, Shahab Sarmashghi, Laksshman Sundaram, et al. Three-dimensional genome landscape of primary human cancers. *Nature Genetics*, pages 1–12, 2025.

[17] Potentiate2024. Study of the CHK1 Inhibitor BBI-355, an ecDNA-directed Therapy (ecDTx), and the RNR Inhibitor BBI-825, in Subjects With Tumors With Oncogene Amplifications (POTENTIATE). https://clinicaltrials.gov/study/NCT05827614, 2024.

[18] King L Hung, Kathryn E Yost, Liangqi Xie, Quanming Shi, Konstantin Helmsauer, Jens Luebeck, Robert Schöpflin, Joshua T Lange, Rocío Chamorro González, Natasha E Weiser, et al. ecDNA hubs drive cooperative intermolecular oncogene expression. *Nature*, 600(7890): 731–736, 2021.

[19] Konstantin Helmsauer, Maria E Valieva, Salaheddine Ali, Rocío Chamorro González, Robert Schöpflin, Claudia Röefzaad, Yi Bei, Heathcliff Dorado Garcia, Elias Rodriguez-Fos, Montserrat Puiggròs, et al. Enhancer hijacking determines extrachromosomal circular MYCN amplicon architecture in neuroblastoma. *Nature communications*, 11(1):5823, 2020.

[20] Biswanath Chowdhury, Kaiyuan Zhu, Chaohui Li, Jens Luebeck, Owen S Chapman, Katerina Kraft, Shu Zhang, Lukas Chavez, Anton G Henssen, Paul S Mischel, et al. Reconstructing the three-dimensional architecture of extrachromosomal DNA with ec3D. *bioRxiv*, pages 2025–02, 2025.

[21] Viraj Deshpande, Jens Luebeck, Nam-Phuong D Nguyen, Mehrdad Bakhtiari, Kristen M Turner, Richard Schwab, Hannah Carter, Paul S Mischel, and Vineet Bafna. Exploring the landscape of focal amplifications in cancer using AmpliconArchitect. *Nature communications*, 10(1):1–14, 2019.

[22] Kevin Hadi, Xiaotong Yao, Julie M Behr, Aditya Deshpande, Charalampos Xanthopoulakis, Huasong Tian, Sarah Kudman, Joel Rosiene, Madison Darmofal, Joseph DeRose, et al. Distinct classes of complex structural variation uncovered across thousands of cancer genome graphs. *Cell*, 183(1):197–210, 2020.

[23] Jens Luebeck, Edwin Huang, Forrest Kim, Ted Liefeld, Bhargavi Dameracharla, Rohil Ahuja, Daniel Schreyer, Gino Prasad, Michał Adamaszek, Rishaan Kenkre, et al. AmpliconSuite: an end-to-end workflow for analyzing focal amplifications in cancer genomes. *bioRxiv*, pages 2024–05, 2024.

[24] Michael J Singer, Larry D Mesner, Cynthia L Friedman, Barbara J Trask, and Joyce L Hamlin. Amplification of the human dihydrofolate reductase gene via double minutes is initiated by chromosome breaks. *Proceedings of the National Academy of Sciences*, 97(14):7921–7926, 2000.

[25] Ofer Shoshani, Simon F Brunner, Rona Yaeger, Peter Ly, Yael Nechemia-Arbely, Dong Hyun Kim, Rongxin Fang, Guillaume A Castillon, Miao Yu, Julia SZ Li, et al. Chromothripsis drives the evolution of gene amplification in cancer. *Nature*, 591(7848):137–141, 2021.

[26] Owen S Chapman, Jens Luebeck, Sunita Sridhar, Ivy Tsz-Lo Wong, Deobrat Dixit, Shanqing Wang, Gino Prasad, Utkrisht Rajkumar, Meghana S Pagadala, Jon D Larson, et al. Circular extrachromosomal dna promotes tumor heterogeneity in high-risk medulloblastoma. *Nature genetics*, 55(12):2189–2199, 2023.

[27] Kaiyuan Zhu, Matthew G Jones, Jens Luebeck, Xinxin Bu, Hyerim Yi, King L Hung, Ivy Tsz-Lo Wong, Shu Zhang, Paul S Mischel, Howard Y Chang, et al. Coral accurately resolves extrachromosomal dna genome structures with long-read sequencing. *Genome Research*, 34 (9):1344–1354, 2024.

[28] Qiuhui Li, Ayse G Keskus, Justin Wagner, Michal B Izydorczyk, Winston Timp, Fritz J Sedlazeck, Alison P Klein, Justin M Zook, Mikhail Kolmogorov, and Michael C Schatz. Unraveling the hidden complexity of cancer through long-read sequencing. *Genome research*, 35 (4):599–620, 2025.

[29] Ayse G Keskus, Asher Bryant, Tanveer Ahmad, Byunggil Yoo, Sergey Aganezov, Anton Goretsky, Ataberk Donmez, Lisa A Lansdon, Isabel Rodriguez, Jimin Park, et al. Severus detects somatic structural variation and complex rearrangements in cancer genomes using long-read sequencing. *Nature Biotechnology*, pages 1–11, 2025.

[30] King L Hung, Jens Luebeck, Siavash R Dehkordi, Caterina I Colón, Rui Li, Ivy Tsz-Lo Wong, Ceyda Coruh, Prashanthi Dharanipragada, Shirley H Lomeli, Natasha E Weiser, et al. Targeted profiling of human extrachromosomal DNA by CRISPR-CATCH. *Nature Genetics*, 54 (11):1746–1754, 2022.

[31] Mădălina Giurgiu, Nadine Wittstruck, Elias Rodriguez-Fos, Rocío Chamorro González, Lotte Brückner, Annabell Krienelke-Szymansky, Konstantin Helmsauer, Anne Hartebrodt, Philipp Euskirchen, Richard P Koche, et al. Reconstructing extrachromosomal DNA structural heterogeneity from long-read sequencing data using Decoil. *Genome research*, 34(9):1355–1364, 2024.

[32] Graham R Brightwell and Peter Winkler. Note on counting eulerian circuits. *arXiv preprint cs/0405067*, 2004.

[33] Peter Ly, Simon F Brunner, Ofer Shoshani, Dong Hyun Kim, Weijie Lan, Tatyana Pyntikova, Adrienne M Flanagan, Sam Behjati, David C Page, Peter J Campbell, et al. Chromosome segregation errors generate a diverse spectrum of simple and complex genomic rearrangements. *Nature genetics*, 51(4):705–715, 2019.

[34] C Passananti, B Davies, M Ford, and M Fried. Structure of an inverted duplication formed as a first step in a gene amplification event: implications for a model of gene amplification. *The EMBO Journal*, 6(6):1697–1703, 1987.

[35] Bonita J Brewer, Celia Payen, MK Raghuraman, and Maitreya J Dunham. Origin-dependent inverted-repeat amplification: a replication-based model for generating palindromic amplicons. *PLoS genetics*, 7(3):e1002016, 2011.

[36] John C Rose, Julia A Belk, Ivy Tsz-Lo Wong, Jens Luebeck, Hudson T Horn, Bence Daniel, Matthew G Jones, Kathryn E Yost, King L Hung, Kevin S Kolahi, et al. Disparate pathways for extrachromosomal DNA biogenesis and genomic DNA repair. *Cancer Discovery*, 15(1): 69–82, 2025.

[37] Laksshman Sundaram, Arvind Kumar, Matthew Zatzman, Adriana Salcedo, Neal Ravindra, Shadi Shams, Bryan H Louie, S Tansu Bagdatli, Matthew A Myers, Shahab Sarmashghi, et al. Single-cell chromatin accessibility reveals malignant regulatory programs in primary human cancers. *Science*, 385(6713):eadk9217, 2024.

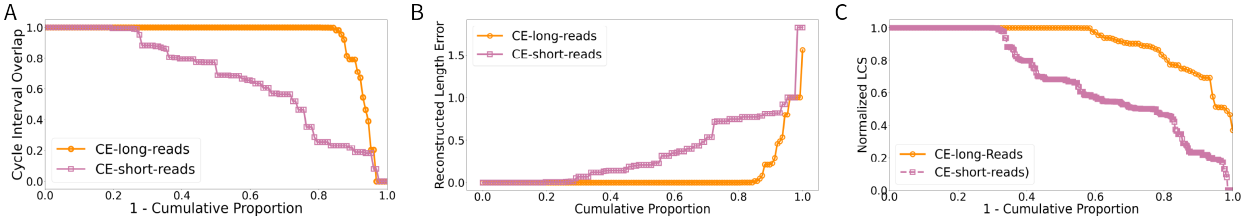
[38] Shuang G Zhao, Matthew Bootsma, Stanley Zhou, Raunak Shrestha, Thaidy Moreno-Rodriguez, Arian Lundberg, Chu Pan, Christopher Arlidge, James R Hawley, Adam Foye, et al. Integrated analyses highlight interactions between the three-dimensional genome and DNA, RNA and epigenomic alterations in metastatic prostate cancer. *Nature genetics*, 56(8): 1689–1700, 2024.

[39] Mahmoud Ghandi, Franklin W Huang, Judit Jané-Valbuena, Gregory V Kryukov, Christopher C Lo, E Robert McDonald III, Jordi Barretina, Ellen T Gelfand, Craig M Bielski, Haoxin Li, et al. Next-generation characterization of the cancer cell line encyclopedia. *Nature*, 569 (7757):503–508, 2019.

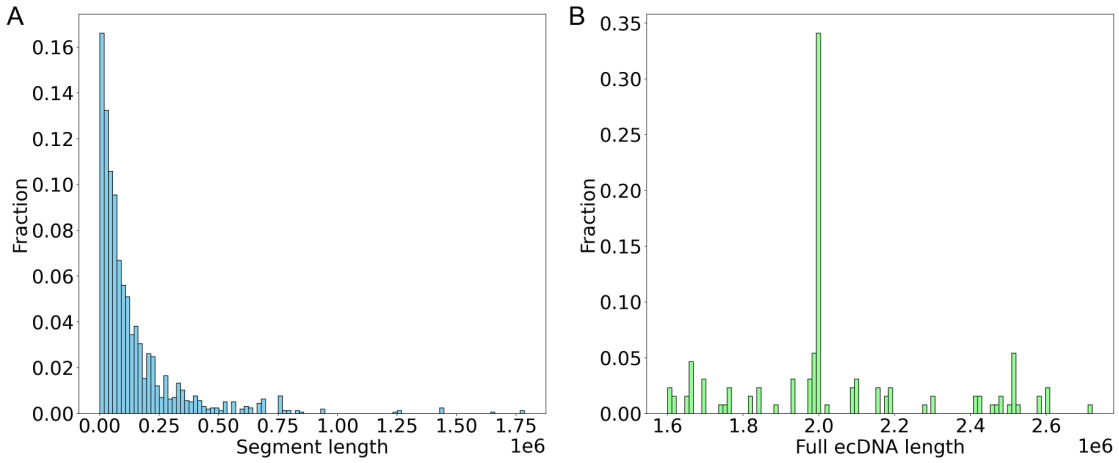
[40] Siavash Raeisi Dehkordi, Ivy Tsz-Lo Wong, Jing Ni, Jens Luebeck, Kaiyuan Zhu, Gino Prasad, Lena Krockenberger, Guanghui Xu, Biswanath Chowdhury, Utkrisht Rajkumar, et al. Breakage fusion bridge cycles drive high oncogene number with moderate intratumoural heterogeneity. *Nature communications*, 16(1):1497, 2025.

[41] Xueyuan Jia, Rongwei Guan, Xiaobo Cui, Jing Zhu, Peng Liu, Ling Zhang, Dong Wang, Yang Zhang, Kexian Dong, Jie Wu, et al. Molecular structure and evolution mechanism of two populations of double minutes in human colorectal cancer cells. *Journal of Cellular and Molecular Medicine*, 24(24):14205–14216, 2020.

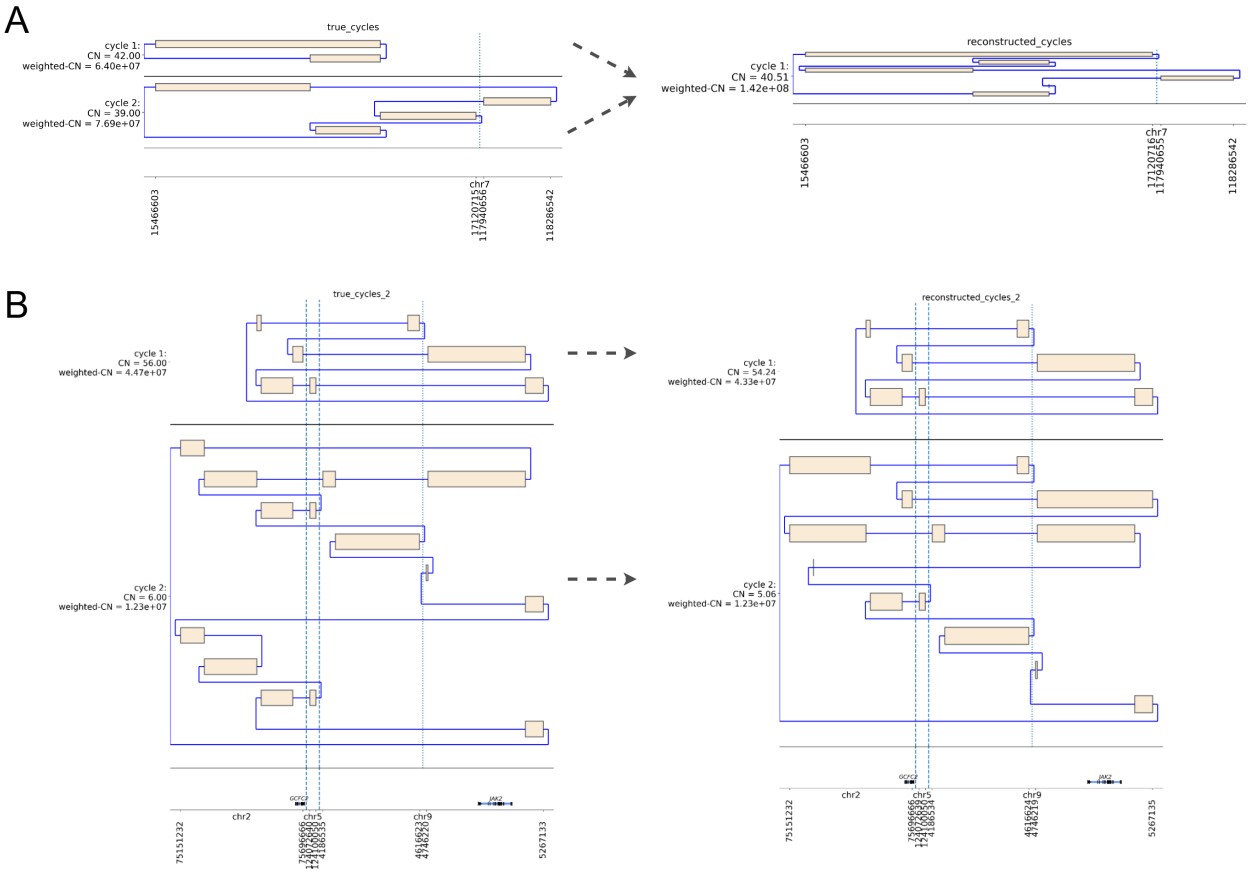
Supplementary Figures



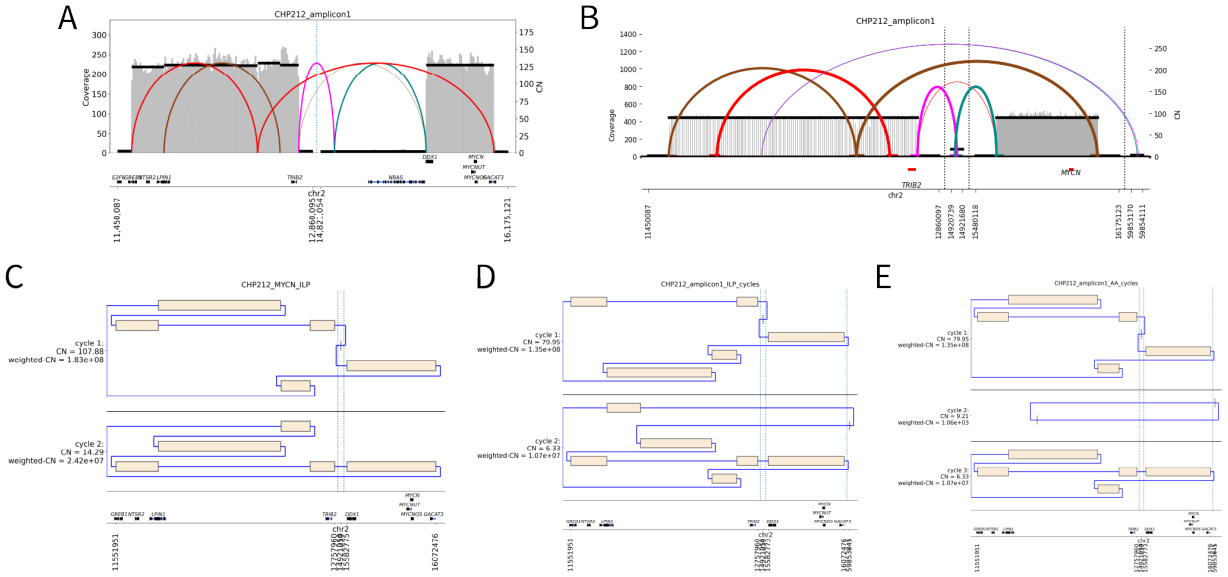
Supplemental Figure S1: **Comparison of A. CIO, B. RLE and C. LCS metrics by CE on simulated data using long-read and short-read graphs.**



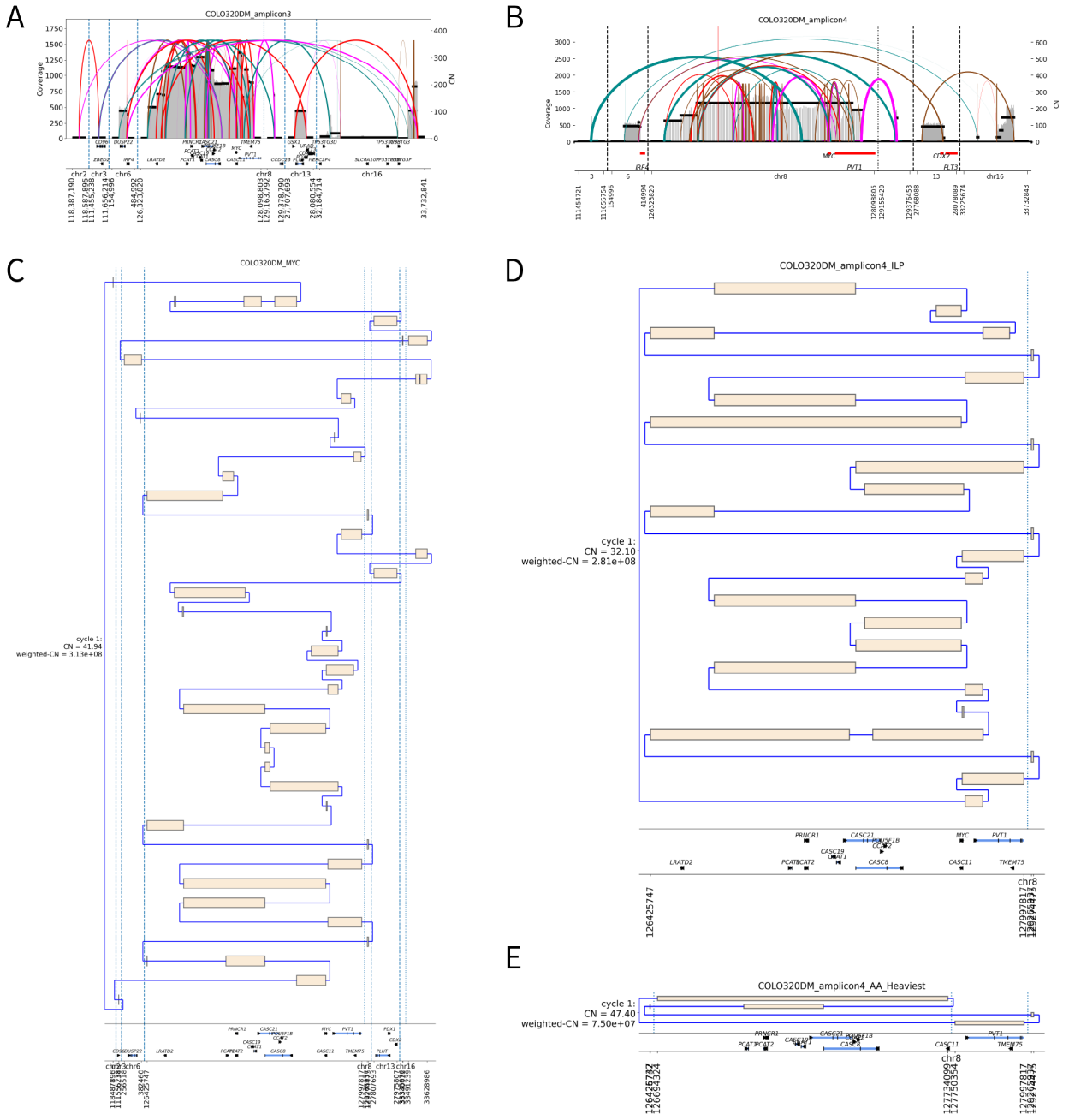
Supplemental Figure S2: **The lengths of the segments versus the lengths of the ecDNA.** The large majority of the segments are smaller than 250K, while ecDNA are larger, with average size exceeding 2Mbp.



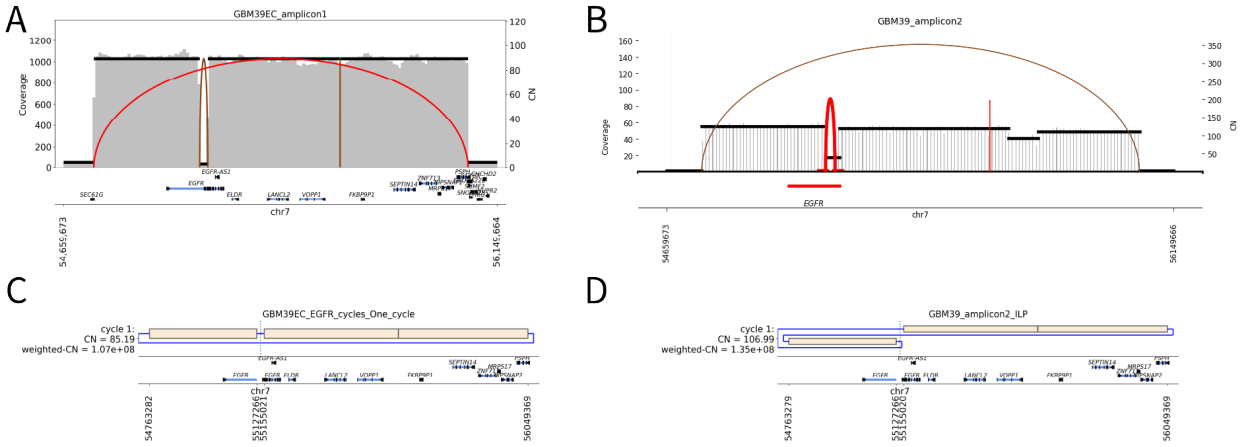
Supplemental Figure S3: **CE behavior on heterogeneous data.** A. CE merges two true cycles onto one when they share genomic segments and are very similar in copy number. B. CE predicts two separate cycles when the true cycles have very different copy numbers.



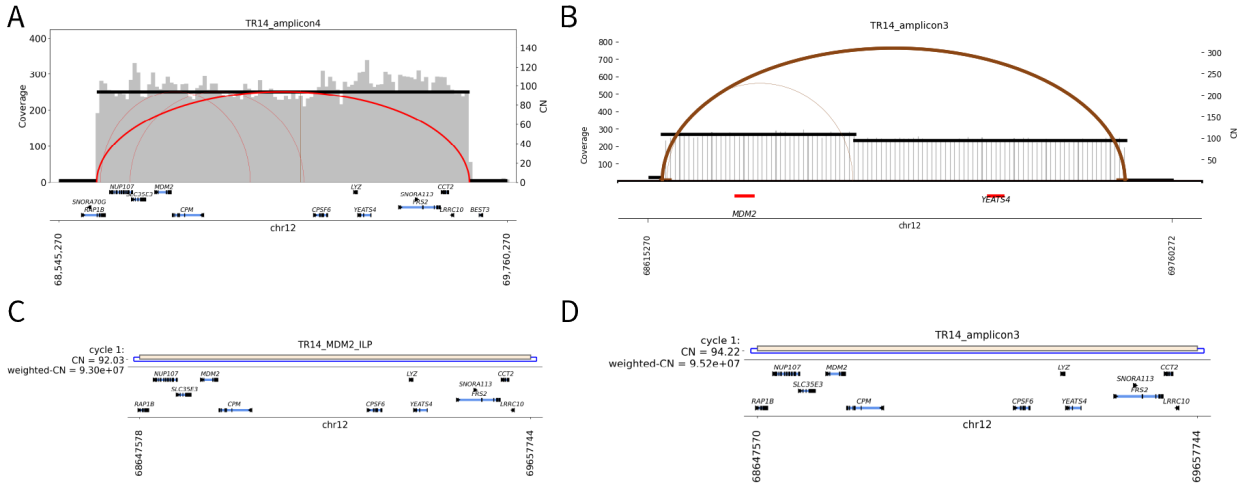
Supplemental Figure S4: **CE reconstruction on CHP-212 cell line.** A. Long read amplicon graph. B. Short read amplicon graph. C. CE reconstruction using long-read graph with 2 cycles, which is identical to CoRAL. D. CE reconstruction using short-read graph. E. AA reconstruction (AA can only extract cycles from short-read graph) with 3 cycles.



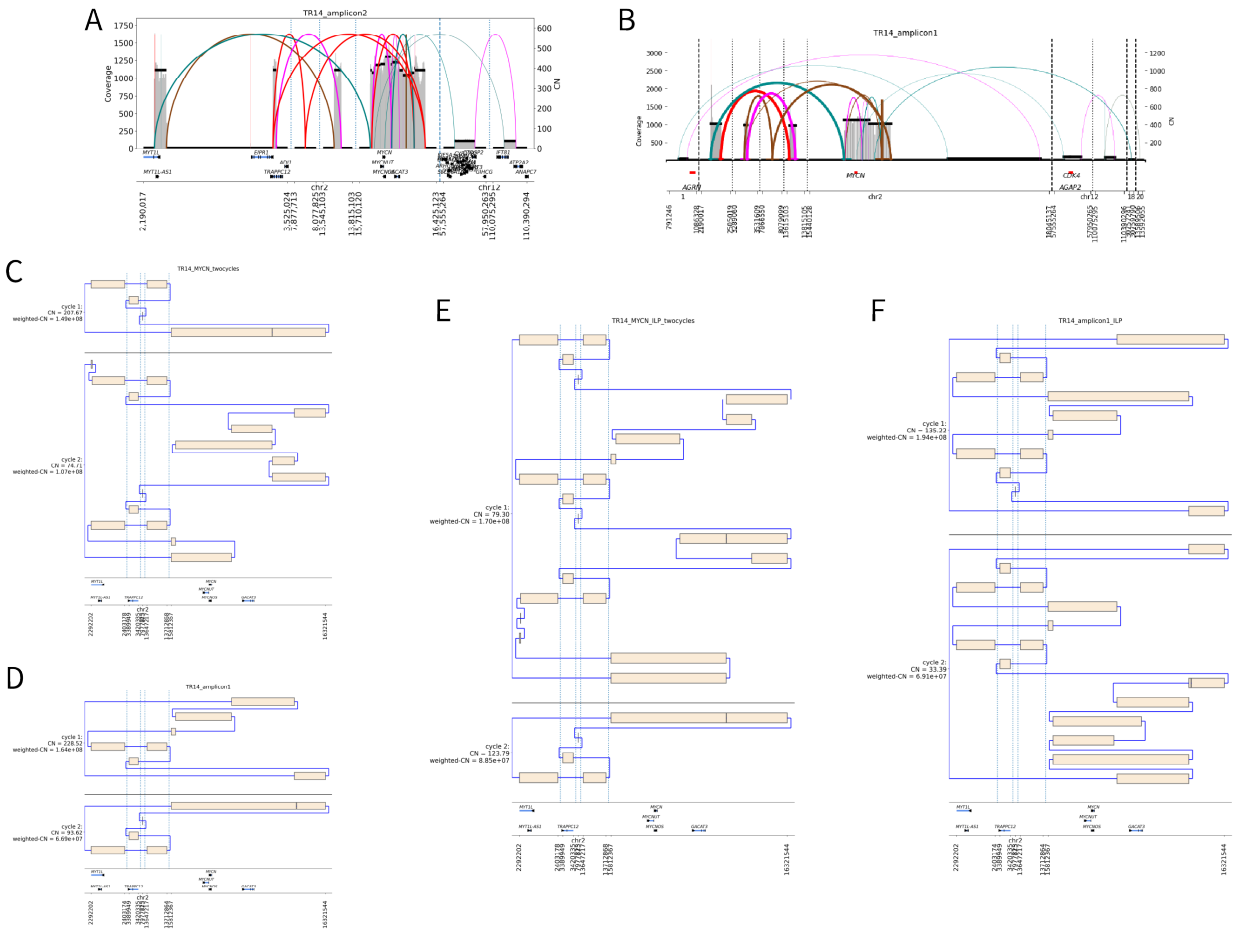
Supplemental Figure S5: **CE reconstruction on COLO320DM cell line.** A. Long read amplicon graph. B. Short read amplicon graph. C. The heaviest cycle from CE reconstruction using the long-read graph, which is identical to CoRAL. D. The heaviest cycle from CE reconstruction using short-read graph. E. The heaviest cycle from AA reconstruction.



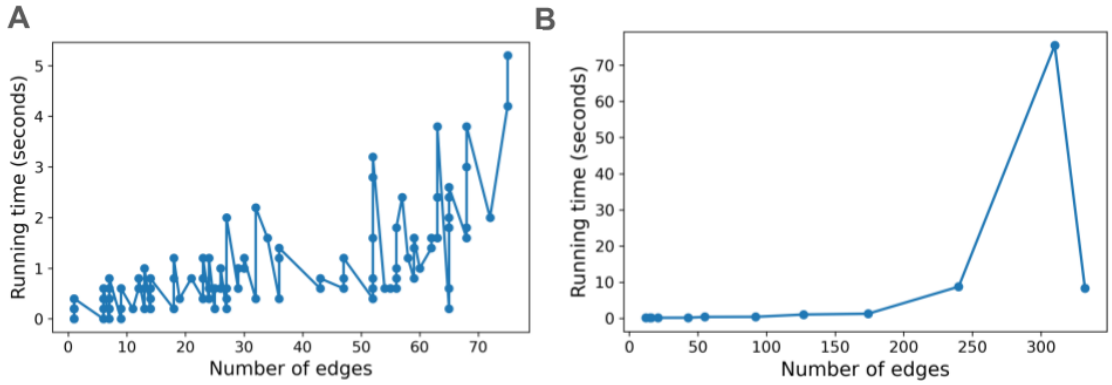
Supplemental Figure S6: **CE reconstruction on GBM39 cell line.** A. Long read amplicon graph. B. Short read amplicon graph. C. The heaviest cycle from CE reconstruction using the long-read graph, which is identical to CoRAL. D. The heaviest cycle from CE reconstruction using the long-read graph, which is identical to AA.



Supplemental Figure S7: **CE reconstruction on TR14 cell line, *MDM2* amplicon.** A. Long read amplicon graph. B. Short read amplicon graph. C. The heaviest cycle from CE reconstruction using the long-read graph, which is identical to CoRAL. D. The heaviest cycle from CE reconstruction using the long-read graph, which is identical to AA.



Supplemental Figure S8: **CE reconstruction on TR14 cell line, *MYCN* amplicon.** A. Long read amplicon graph. B. Short read amplicon graph. C. 2 heaviest cycles from CoRAL reconstruction. D. 2 heaviest cycles from AA reconstruction. E. 2 heaviest cycles from CE reconstruction, using long read graph. F. 2 heaviest cycles from CE reconstruction, using short read graph.



Supplemental Figure S9: **Running time vs the number of edges of the graph.** A. CE running time vs number of edges for simulated long-read data from cell lines. B. CE running time for long-read (ONT) data on cell lines.

S1 Supplementary Methods

S1.1 Full ILP for the *optimization* step

The full CE MILP formulation is provided below.

Parameters.

M**:** maximum multiplicity allowed for concordant and discordant edges

$\gamma(W)$**:** $= 0.01 \sum_{(u,v)\in W} C_{uv}\ell_{uv}$, and balances weights of subwalk constraints relative to LWCN in the objective.

$p^j_{uv} \in \{0,1\}$: indicates if edge (u,v) is in subwalk p_j

Variables.

$F \in \mathbb{R}$: copy number assigned to edges in the solution but with multiplicity one.

$x_{uv} \in \{0,1\}$: $x_{uv} = 1$ if edge (u,v) is selected, and 0 otherwise.

$f_{uv} \in \mathbb{R}$: copy number assigned to edge (u,v) in the solution, should be multiples of F.

$y_{uvm} \in \{0,1\}$: (auxiliary) $y_{uvm} = 1$ iff (u,v) has multiplicity $m \in \{1 \le m \le M\}$ for each discordant edge in a feasible walk.

$z_{uvm} \in \mathbb{R}$: The auxiliary variable is used to constrain $z_{uvm} = m \cdot F \cdot y_{uvm}$ which is a quadratic formula but is linearized later.

$P_j \in \{0,1\}$: indicates if subwalk p_j is satisfied $1 \le j \le p$.

$d_u \in \{0,1,2,..\}$: indicates the order of the nodes, used in connectivity constraints.

$c^+_{uv} \in \{0,1\}$: be a binary variable that is 1 if the direction is from u to v, again used in connectivity constraints.

CE MILP. CE maximizes the sum of LWCN and γ-weighted subwalk constraint using

$$\max \sum_{(u,v)\in E} f_{uv}\ell_{uv} + \gamma \frac{\sum_j P_j}{|\mathcal{P}|}$$

while satisfying the following constraints.

1. **capacity**: For any edge (u,v) in a feasible solution, its copy number (f_{uv}) must not exceed its capacity (C_{uv}). If the edge is not chosen in the solution, its copy number is zero.

$$f_{uv} \le C_{uv} \cdot x_{uv} \;\; \forall (u,v) \in E$$

2. **balance**: The sum of the solution copy number of discordant and concordant edges that connect to a node is equal to the solution copy number of the sequence edge connected to that node.

$$\sum_{(w,u)\in E_c \cup E_c} f_{wu} = \sum_{(u,v)\in E_s} f_{uv}$$

3. **copy number F with multiplicity one**: Note that if edge (u,v) is picked in the solution (i.e., $x_{uv} = 1$), then $F \le f_{uv}$, but otherwise (u,v) imposes no constraint. For this purpose we introduce a sufficiently large constant $C_{\max}$ (e.g., $C_{\max} = M \cdot \max_{(u,v)\in E} f_{uv}$).

$$F \le f_{uv} + C_{\max} \cdot (1 - x_{uv}) \;\; \forall (u,v) \in E_c \cup E_d$$

4. **discordant edge multiplicity constraint**: The multiplicity of a discordant edge is managed using

$$\sum_{m=1}^{M} y_{uvm} = x_{uv} \qquad \forall (u,v) \in E_d.$$

Subsequently the copy of number of the edge is given by the (quadratic) constraint

$$f_{uv} = \sum_{m=1}^{M} m \cdot F \cdot y_{uvm} = \sum_{m=1}^{M} z_{uvm} \qquad \forall (u,v) \in E_d$$

which guarantees f_{uv} are multiples of F. We linearize the quadratic constraints using

$$\begin{aligned} z_{uvm} &\leq mF \\ z_{uvm} &\leq C_{\max} \cdot y_{uvm} \\ z_{uvm} &\geq mF - C_{\max} \cdot (1 - y_{uvm}) \\ z_{uvm} &\geq 0 \end{aligned}$$

5. **subwalk constraints**: If a subwalk is in the solution ($P_j = 1$), all of its edges are in the solution. If a subwalk in not in the solution ($P_j \neq 0$), there is not constraint here.

$$x_{uv} \geq P_j \cdot p^j_{uv} \qquad \forall j \in \{1, ..., p\} \qquad \forall (u,v) \in E \cap p_j$$

6. **connectivity constraints 1 (start node):** Assign s to be the start node.
The order of s is 1.

$$d_s = 1$$

No direction into s.

$$\sum_v c^+_{vs} = 0$$

Direction out of s

$$c^+_{sv} \leq \sum_v x_{sv}$$

7. **connectivity constraints 2 (order of nodes):** If node u is not in the solution, then $d_u = 0$, otherwise d_v is a positive integer which reflects the order of the node.

$$d_u \leq |V| \sum_v x_{uv} \qquad \forall u \in V$$

8. **connectivity constraints 3 (edge direction):** Only one direction from each edge in the solution

$$c^+_{uv} + c^+_{vu} \leq x_{uv} \qquad \forall u, v \in V$$

9. **connectivity constraints 4 (incoming edge for nodes):** If a node is part of the solution, and it is not s (the starting node), then it must have at least one incoming direction

$$\sum_v c^+_{vu} \geq \frac{1}{|V|} \sum_v x_{uv} \qquad \forall u \in V - \{s\}$$

10. **connectivity constraints 5:**

$$d_v - d_u \geq 1 - |V|(1 - c^+_{uv}) \quad \forall u, v \in V$$

S1.2 CE vs CoRAL

In this section we summarize and compare the formulations of CoRAL and CE, see Supplementary Table S1. In CoRAL (specifically the max weight formulation), the objective is quadratic and there are 20 classes of constraints, where 4 of them are quadratic and the others are linear. We converted them into 16 classes of MILP constraints. One key difference between CE and CoRAL is how edge multiplicity is obtained. In CE, we do not introduce an explicit multiplicity variable for edges. Instead, the model includes a variable representing the copy number of the ecDNA cycle, along with variables representing the copy number contributed by each edge. The multiplicity of an edge can then be calculated after solving the MILP optimization by dividing the edge copy number by the copy number of the ecDNA cycle. In contrast, CoRAL directly models edge multiplicities as decision variables within the optimization framework.

Model	**CE**	**CoRAL**
Objective	Linear	Quadratic
Eulerian constraint: cycle/walk(s) can be traversed by alternating sequence and breakpoint edges	Linear	Linear
CN Constraint: total CN of edge e in the cycle/walk $\leq$ remaining CN(e) $\forall$ edges e	Linear	Quadratic
Multiplicity constraint: repeat times of edge e $\leq R(e)$ for each discordant edge e	Linear	Linear
Subwalk constraints	Linear	Linear
Connectivity constraint	Linear (optional)	Quadratic

Supplementary Table S1: Comparison of model formulations for CoRAL and CE.

S1.3 Integer multiples of F of all edges in the solution

Lemma S1.1. *If the copy numbers of the discordant edges in the cycle are positive integer multiples of F (as enforced by Constraint 4), then the copy numbers of the concordant and sequence edges in the solution cycle are also positive integer multiples of F.*

Proof. We consider a canonical ordering of all nodes in the MILP solution based on their chromosomes and then on their nucleotide coordinates. By definition, node 1 in the solution is connected to a sequence edge and one or more discordant edges, but not to any concordant edge. We prove by induction that for each node k in the ordering,any edge incident on node k must have a copy number that is an integer multiple of F.

Base case ($k = 1$). We know that node 1 is only incident on a single sequence edge, and all other edges connecting to 1 are discordant, with copy numbers that are multiples of F. By the balance condition, the sequence edge $(1, 2)$ must also have a copy number that is an integer multiple of F. Therefore, all edges connected to node 1 have copy numbers that are positive integer multiples of F.

Induction hypothesis. Assume that every edge incident to nodes $1, 2, ..., k$ has copy number equal to a positive integer multiple of F.

Inductive Step. We now consider node $k + 1$. All discordant edges incident on node $k + 1$ must have copy numbers that are multiples of F, and we need to investigate the concordant and sequence edges (denoted e_c, e_s, respectively) incident on $k + 1$. We have two cases.

First, suppose $\mathrm{CN}[e_c] = 0$. Then, by the balance condition, $\mathrm{CN}[e_s]$ must be an integer multiple of F. Second, we consider the case $\mathrm{CN}[e_c] > 0$, then both nodes must be a part of the solution, and by our ordering criteria, $e_c = (k, k+1)$, connects node k to $k+1$. Then, using induction hypothesis, $\mathrm{CN}[e_c]$ must be an integer multiple of F, and by balance, $\mathrm{CN}[e_s]$ must be a positive integer multiple of F.

□

S1.4 Implementation details

Preprocessing amplicon graphs. While CE accepts an amplicon graph directly from the output of AA[21] or CoRAL[27], it replaces each foldback edge $(u, u) \in E_d$ with $(u, u_1) \in E_d$, $(u_1, u_2) \in E_s$, and $(u_2, u) \in E_d$, and setting $\ell_{u_1 u_2} = 1$. This construction allows alternate paths in the model to traverse through foldbacks and, by assigning a positive length, encourages their inclusion in the solution based on the objective function.

Connectivity enforcement cycle extraction. When CEc aims to extract a cycle (to avoid multiple disconnected cycles), it connects artificial source (s) and sink (t) nodes to all nodes in the graph. To extract exactly one cycle, CEc finds a path from s to t, with s and t connected to the same node in the solution. To ensure feasibility with respect to the alternating edge constraints, s is connected to other nodes via discordant edges, while t is connected via sequence edges. The copy number assigned to each of these newly added edges equals the maximum copy number of the edges originally adjacent to that node. The length of the sequence edge connecting t is set to 1 - positive, but short - so that it encourages selection in the solution.

Extract paths from graph. The CE(c) method is an iterative framework that, in each iteration, takes as input an amplicon graph with copy number annotated edges. It formulates and solves a mixed-integer linear programming (MILP) problem to identify a subset of edges, each with an assigned copy number, that can form a cycle. After each iteration, the graph is updated by subtracting the copy numbers of the selected edges from their corresponding values in the graph. The updated graph is then used as input for the next iteration to extract additional cycles. Although CE/CEc can also extract paths, it prioritizes the detection of cycles. Once no cycles remain, the algorithm proceeds to extract paths.

If CE/CEc aims to extract a path after all cycles have been extracted, it introduces artificial source (s) and sink (t) nodes. Both nodes are connected to all other nodes in the graph via discordant edges, and each new edge is assigned a copy number equal to the maximum copy number of the edges originally adjacent to that node.

S1.5 Cell lines used in this paper

For building the amplicon graph, we used the following: AmpliconArchitect (AA) version 1.5.r1; CoRAL version 2.2.0.

Cell line	Source	Genes	ONT	Illumina
TR14	Hung 2021[18]	*MDM2*	✓	✓
TR14	Hung 2021[18]	*MYCN, CDK4*	✓	✓
TR14	Hung 2021[18]	*ODC1*	✓	
TR14	Hung 2021[18]	*SMC6*	✓	
HK296	Turner 2017[1]	*MDM4*		✓
HK296	Turner 2017[1]	*EGFR*		✓
GBM39	Turner 2017[1]	*EGFR*	✓	✓
GBM39_mono	Zhu 2024[27]	*EGFR*	✓	
SF295	Turner 2017[1]	*CCND1*		✓
H460	Turner 2017[1]	MYC		✓
MB411FH	Turner 2017[1]	*MYC*		✓
CA718	Turner 2017[1]	*MDM2, MYCN, PDGFRA*		✓
CHP-212	Ghandi 2019[39]	*MYCN*	✓	✓
UACC1598	Luebeck 2024[23]	*MYCN*		✓
CCFSTTG1	Luebeck 2024[23]	*CDK6, MDM2*		✓
HL-60	Turner 2017[1]	*MYC*		✓
NCIH2170	Dehkordi 2025[40]	*MYC*	✓	✓
MB002	Turner 2017[1]	*MYC*		✓
HK301	Turner 2017[1]	*EGFR*		✓
H524	Luebeck 2024[23]	*MYC*		✓
NCIH69	Ghandi 2019[39]	*MYCN*		✓
HK359	Turner 2017[1]	*EGFR*		✓
GBM6	Turner 2017[1]	*EGFR*		✓
DMS273	Ghandi 2019[39]	*MYC*		✓
DMS273	Ghandi 2019[39]	*MYCL, ARNT*		✓
COLO320DM	Zhu 2024[27]	*MYC*	✓	✓
COLO320DM_Hung2021	Hung 2021[18]	*MYC*	✓	
H716	Jia 2020[41]	*MYC, FGFR2*		✓
SCLC21H	Ghandi 2019[39]	*MYC*		✓
PC3	Turner 2017[1]	*MYC*	✓	✓
SNU16	Hung 2022[30]	*MYC, FGFR2*	✓	

Supplementary Table S2: **Amplicons used for testing cycle extraction.** Each row describes a unique amplicon defined by its amplicon graph.