Simulating Balanced Allocation of Project Float to the Critical Path in Network Schedules

Gunnar LUCKO¹, Richard C. THOMPSON, Jr. ², and Yi SU³

¹ Construction Engineering and Management Program, Department of Civil Engineering, Catholic University of America, 620 Michigan Avenue NE, Washington, DC 20064; phone 202-319-4381; fax 202-319-6677; email: lucko@cua.edu, thompsrc@cua.edu, and 61su@cardinalmail.cua.edu.

ABSTRACT
Recent work suggests replacing the unresolved construction scheduling question ‘who owns float’ by ‘how can float reduce the risk level across a project’. In other words, how to best distribute the time from project finish based just on productivities and the contractual finish? First, critical activities: They form the longest continuous path through a schedule and by definition are at risk. Delays to them will affect the performance of the entire project. This research assesses conceptual shortcomings of the related critical chain project management, and develops an algorithm that simulates how float can be allocated to activities by considering their individual risk. It quantifies how much sampled durations exceed their individual allocation. It is not intended to replace the experience of construction managers who empirically assign buffers to risky activities, but augment it so that an optimized float allocation can be determined for given activity risk. Allocation models build upon Penrose and examine exponents zero and one; ultimately validating the square root. Its contribution is an ability to quantify how schedules can be effectively protected.

INTRODUCTION
In light of frequent delays that inundate construction projects of many types and sizes (e.g. Bhargava et al. 2010, Acharya et al. 2006), reaching double-digit percentages of the originally planned durations, project management still requires better preventive scheduling. Such approach should minimize delays proactively or, if they occur, have provisions in place to mitigate then. Existing attempts sought to accomplish this via float or buffers and are reviewed below. They provide flexibility in terms of time to a construction schedule. But a tradeoff problem exists (van de Vonder et al. 2005); too little flexibility may be insufficient to protect against delays, but too much may become wasteful, as proponents of the lean construction philosophy warn. Their nature, features, and appropriate sizing and placement should therefore be explored.

Research Goal and Objectives
The goal of this research is to overcome the notion of the critical path itself in what Lucko and Thompson (2015) called a ‘criticality paradox’; i.e. prevalent construction
scheduling creates a ‘two class society’ of critical activities and those that happen to have received float as an artifact of composing the schedule. But schedules should be protected so that their activities have a resistance that is appropriate to their individual risk within the schedule. Float or buffer strategies should be preventive and applied \textit{a priori}, so that clear rights and responsibilities can be derived. Such transparency can also reduce legal claims. Objectives for this exploratory research are set as follows:

1. Extract conceptual limitations of float and buffer management approaches;
2. Derive an algorithm that improves them toward a viable mitigation technique;
3. Validate the new approach by testing complex schedules through simulation.

FLOAT OWNERSHIP AND BUFFER MANAGEMENT
External or internal events can cause disruptions that negatively impact schedules due to their dependency structure. A predecessor activity with late start or longer duration can cause a ripple effect that passes through a schedule (Lucko and Thompson 2015). This may prevent its successors from starting on time, inconveniences or harms their respective subcontractors, or even worse, pushes the project finish over its contractual deadline. Protection mechanisms are thus vital. Buffers (operations research) and float (construction management) in schedules are two closely related phenomena that only differ in whether they are intentional. The former are planned; as “buffers operate to provide a cushion or shield against the negative impact of disruptions and variability” (Russell \textit{et al.} 2014b). The latter are artifacts of how the activity durations and links between them compose a schedule. This distinction created diverse research: Studies in construction focused on float ownership, a contentious issue with many existing contradictory opinions (Al-Gahtani 2009): Some argued that the owner’s proprietor role lets them decide on float use; others that the general contractor’s (GC) executive role is in charge of a schedule. But they only studied ‘passive’ float, not ‘active’ buffers, measured it only within a baseline schedule, but not beyond its finish to the contract deadline, and failed to consider subcontractors who perform actual activities.

Critical Chain Project Management
The most noteworthy attempt to protect schedules from delays exists in critical chain project management (CCPM) (Goldratt 1997), which is the implementation of the so-called Theory of Constraints, a production philosophy that Goldratt (1984) novelized. It advocated a continuous improvement process with rigorous prioritization: Examine a manufacturing system, identify the most severe bottleneck that hinders throughput, assign more resources or resolve otherwise, and repeat for next constraint. Extending it to CCPM created a project view, instead of just locally “adding safety at each step, most of which is wasted” (Tukel \textit{et al.} 2006, p. 402). But excessive duration estimates “may lead to a so called “student syndrome” – a situation when a contractor … for a particular task waits until the last possible moment to start” (Jaśkowski and Biruk 2011). Such procrastination is also called Parkinson’s Law (Russell \textit{et al.} 2014a). The reasonable basic idea of CCPM is thus to insert buffers only where they are necessary and size them to be useful. How it performs on these two items will determine its success. The different types are \textit{project (PB), feeder (FB), and resource buffers (RB)}.

But CCPM rests on severe assumptions: 1. ‘Safe’ durations are viewed as inflated from 50% success probability to nearly 100% (Herroelen and Leus 2001) and halved
before scheduling them (Tukel et al. 2006). 2. The baseline schedule places activities 
as late as possible (Herroelen and Leus 2001). 3. Similar to the critical path method 
(Wiest 1964), the longest path through a network is found, but called critical chain. 4. 
Laborers may multitask on several projects; thus resource leveling is employed until a 
feasible baseline schedule is found. 5. Time that was cut aggressively from the critical 
chain is halved again and aggregated into a project buffer after the last activity (Tukel 
et al. 2006). 6. To protect the critical chain against disturbances, feeder buffers are 
inserted wherever side paths merge into it. Their length is “half the duration of the 
non-critical chain path leading into it” (Herroelen and Leus 2004, p. 1604). 

Many weaknesses exist in CCPM: 1. Theory of Constraints focuses on a single 
obstacle, which is narrower than the ‘80/20’ Pareto rule-of-thumb. 2. For its resource 
leveling, many methods and heuristics exist that may give different results. 3. Timing 
activities as late as possible is not proactive, but discourages early completion. 4. If 
more than one critical chain exists, one is arbitrarily selected (Herroelen and Leus 
Propagating any delay directly into the project buffer makes it harder to trace to its 
root cause. 7. Based on assuming only 50% success probability, consuming buffers is 
accepted, but would lead to the unfair ‘first-come, first served’ policy. The original 
CCPM does not provide any explicit buffer ownership allocation. Al-Gahtani (2009, 
p. 88) suggested that “the party who has the greatest risk… should be entitled to float 
ownership and deserves compensation from other … parties who increase the risk 
associated with the project by consuming the float”, but CCPM lacks such provision.
8. It is unknown if initial durations were doubled. The “50% rule … may lead to a 
serious overestimation of the required buffer protection” (Herroelen and Leus 2001, 
p. 559), because a linear relation to time does not optimally temper risk. 9. Its ‘25% 
of safe’ PB size is just a rule-of-thumb. 10. Its ‘25% of safe’ FB size is a rule-of-
thumb; alternatives were proposed: “Newbold (1998) recommends … two times the 
standard deviation of the feeding chain based on the assumption that the sum of the 
durations … has a normal distribution (Tukel et al. p. 407). 11. Feeder buffers may 
even push non-critical paths to before the start of the critical chain. 12. Activity dates 
are less important than the project, as “Goldratt (1997) and Newbold (1998) argue in 
favor of the … relay race..., i.e. start an activity as soon as the predecessor activities 
are finished” (Herroelen and Leus 2001, p. 566). But this contradicts the latest start 
baseline. One study opposed it in that a “project manager schedules tasks according 
to the planned times rather than following a “roadrunner mentality” ” (Tukel et al. 
2006, p. 408). 13. Resource buffers act as mere signals that indicate a transition of 
type only and have no impact upon the schedule dates (Herroelen and Leus 2001).

Vanhoucke (2012) listed four methods to size feeder buffers only. ‘Cut and paste’ 
merely cuts durations on a feeder path in half and pastes half of that at its merge into 
the critical chain. ‘Root square error’ treats half of the squared difference of average 
and safe duration as the variance of activity duration (assuming durations to be non-
skewed, more simplistic than PERT), adds these values on a feeder path, and takes its 
square root as the feeder buffer (Newbold 1998). ‘Adaptive’ methods used a network 
metric: For ‘density’, one plus the coefficient of network complexity was multiplied 
by said square root to gain the feeder buffer. For ‘resource tightness’, said factor was 
replaced by the so-called restrictiveness estimator (cf. next section, Tukel et al. 2006).
Network Schedule Simulations

Several CCPM-related studies examined the efficacy of buffer management policies via computer simulation and related aspects. Before any simulations were possible, a varied set of networks was needed. Several random network generators were created, e.g. for activity-on-arrow networks (Demeulemeester et al. 1993), DAGEN (Agrawal et al. 1996), ProGen (Kolisch and Sprecher 1996), RiskNet (Tavares et al. 2002), and RANGEN (Demeulemeester et al. 2003). Their randomness was compared and found to depend on how networks are built: Add links to a bare schedule or delete from full linked one; and how duplicates are identified via a dependency matrix (Vanhoucke et al. 2008). Their output were networks with a user-defined target complexity, but only finish-to-start links without leads or lags. Many complexity measures were created to characterize the topology of schedules via the respective nodes and links between them (e.g. Tavares et al. 1999, Nassar and Hegab 2006). Common measures were the coefficient of network complexity (CNC); complexity index (confusingly abbreviated CI like the criticality index) for activity-on-arc networks (Vanhoucke et al. 2008); and order strength (OS, density) – in order of their performance quality from low to high (De Reyck and Herroelen 1996); and also the restrictiveness estimator (RT):

\[
\text{CNC} = \frac{\text{Actual number of links}}{\text{Actual number of nodes}} \\
\text{OS} = \frac{\text{Actual number of links}}{\text{Possible number of links}}
\]

Where possible number of links among n non-dummy activities (no zero duration project start and finish) is \( n \cdot (n - 1) / 2 \) (Demeulemeester et al. 2003). The CI is given by an algorithm that counts the steps “to reduce a two-terminal acyclic network to a single edge” (Demeulemeester et al. 2003, p. 19). The RT requires more; it requires a ‘reachability matrix’ (Schwindt 1995) that is derived from the precedence matrix.

Benchmark sets of schedules were needed for researchers to consistently compare their studies, e.g. Kolisch and Sprecher’s (1996) Project Scheduling Problem Library (PSPLIB). It contains random sets for the resource-constrained project scheduling problem, plus sets with minimum / maximum lags or multiple modes (i.e. different possible crews or production rates) with optimal or heuristic solutions (if known) or bounds. Its 480 or 600 schedules contain 30, 60, 90, or 120 non-dummy activities.

Construction Schedule Simulations

Park and Peña-Mora (2004) reviewed Goldratt (1997) and some lean construction studies, but seemed unaware of the aforementioned network studies from operations research. They criticized mere percentage-based contingency for encountering the student syndrome and that the CCPM project buffer was reactive, not proactive. But their premise was similar to CCPM, treating surplus time beyond average activity duration as potential contingency (instead of halving durations). It was placed “in front of the successor… to find problems made in the predecessor” and get ready by “ramping up” for its successor (ibid, p. 628). Different from CCPM, activities were linked to overlap: The average predecessor finish was set concurrent to its successor ‘reliability buffer’ start. Buffer sizing claimed to reflect relative productivities, the “robustness against uncertainties”, and “[s]ensitivity… to changes” (ibid, p. 62), but resorted to assigning simple multipliers. The ‘dynamic’ approach simply updated link lag and reliability buffer size while a predecessor was ongoing, both if it was faster or
slower than planned, but lacked details. Unlike CCPM, it allowed four link types. Park and Peña-Mora (2004, p. 627) simulated a small system dynamics model that “pools, relocates, resizes, and recharacterizes any [percentage] contingency buffer”. Only two activities with custom buffer versus equal or no buffer were compared. A brief calculation of 28 activities with unclear variability appeared to support buffers.

Buffering in construction projects was explored further by Barraza (2011), who also criticized the subjective current practice that just adds an empirical percentage to durations. He performed 10,000 Monte Carlo simulation runs of a bridge project. To represent uncertainty from risk, its 32 activities had uniform, normal, or symmetric or right-skewed triangular distributions, inspired by the Program Evaluation and Review Technique (PERT). Integer durations were sampled. The ‘planned’ project duration including contingency was set so that it was exceeded in only 20% of simulation runs. A critical path was composed apparently by the highest values of the criticality index (CI)—the percentage of runs in which it was critical—of each probabilistic activity that formed a continuous path through the network. Adding the medians of simulated durations of these critical activities (again from the simulation runs) gave a ‘target’ project duration. To prevent a ‘first-come first-serve’, contingency (the difference of planned minus target project durations) was distributed as follows: For all activities on the critical path, the same unknown percentile on their probability distributions was iteratively varied so that the sum of their durations at said percentile equaled the planned project duration. This was the ‘percentile’ duration. Said percentile durations minus median durations were activity contingencies, whose sum was the correct total.

That study inserted contingency on the critical path. But the vital question of float ownership was unaddressed. Herroelen (2013) criticized more issues: 1. Critical chain also used median durations, but they are not statistically valid to add on a path; means are. 2. Monte Carlo already exists in commercial scheduling software. 3. Three-point estimates to represent risk are simplistic, because risk occurs randomly, not according to a concave distribution. 4. Using percentile iterations is cumbersome. 5. Resource limits were ignored, but other approaches at least created resource-feasible schedules.

This extraction of existing conceptual limitations fulfills Research Objective 1.

METHODOLOGY
Due to these conceptual limitations, a new approach is sought. Thompson and Lucko (2011) introduced the idea float could be explicitly allocated to critical activities as inspired by decision-making in other areas, particularly voting power (Banzhaf 1965). Summarized briefly, it has been shown that the potential impact of individual actions (vote) on a group (election) depends not on the size of the respective individual, but on the number of times where said individual’s action determines the group outcome. In turn, derivations of the frequency of such dominance found that these are related to the square root of the size (Penrose 1946). Based on this insight, this research will use a proxy measure of risk as ‘size’ of each activity, which could be its duration, cost, a risk factor, or a weighted combination of such items into a single numerical quantity. It is hypothesized that this square root (size to the power of one half) could strike a balance between the extremes ‘equal float for all’ (size to power of zero) or ‘float in direct proportion’ (size to power of one). Lucko and Thompson (2015) list the details.
The previously mentioned limitation of network generators and benchmark sets containing only finish-to-start links can be overcome by converting such other links (start-to-start, start-to-finish, and finish-to-finish) into equivalent finish-to-start ones that place activities identically. This may include lead or lag durations. It is assumed that schedules do not contain redundant links (e.g., for an activity sequence A → B → C, a link A → C would be redundant or transitive), otherwise they would need to be removed first. Future research may extend the approach to allow such complications.

The new methodology for float allocation is envisioned to function as follows: Create a schedule with known probability distributions. Assume that it contains no dummy links. Prior resource leveling is omitted in this study for brevity. Identify the critical path for productivity-based (i.e., assumed uninflated) durations. Contract float (CF) will be set as a yet-to-be-specified period beyond the ‘raw’ project finish, whose ownership will be allocated into distributed float (DF). Next, simulate said schedule repeatedly to obtain the criticality index of each activity. It encompasses the impact within the network as how often said activity may become critical (and thus prone to cause problems if delayed). It also somewhat reflects the complex network topology that influences its behavior. Risk is captured by the proxy of duration, because longer activity have more opportunity to incur delays. (Note that more factors like technical difficulty or cost could be built into a weighted risk value.) Together, they reflect the definition: Risk is the probability of a negative event times the impact if it actually occurs. Each critical activity now receives DF ownership with a weight of the square root of the product of its duration and criticality index. Next, in a second round of simulation, it is measured how often and to what extent each activity would exceed its personal allocation. A relay policy is assumed, i.e., if an activity can start, it should. This is realistic, because in practice, subcontractors are expected to move in quickly. This approach is suitable for optimization, because it merely assigns a percentage of ownership and does not modify the ‘raw’ schedule, so that CF can still be varied.

Table 1 compares the CCPM step with the envisioned new allocation approach. This proposed approach for a proactive float allocation fulfills Research Objective 2.

<table>
<thead>
<tr>
<th>Critical Chain</th>
<th>Proposed Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latest start schedule with ‘safe’ durations (assumed as inflated)</td>
<td>Earliest start schedule with productivity-based durations, assign activity duration distributions</td>
</tr>
<tr>
<td>Halve durations, resource-level, identify longest continuous sequence as critical chain</td>
<td>Resource-level, identify sequence with zero float as critical path, which may branch or merge</td>
</tr>
<tr>
<td>Project buffer (PB) after last activity equals half of time cut previously from critical chain (Not applicable.)</td>
<td>Available contract float (CF) after last activity is contract finish minus productivity-based project finish; simulation determines actual needed CF</td>
</tr>
<tr>
<td>No individual PB ownership, but ‘first-come first-serve’</td>
<td>Monte Carlo simulation to calculate criticality index, possible use it in weighted risk factor</td>
</tr>
<tr>
<td>Insert feeder buffers (FB) of half ‘safe’ side path length wherever it</td>
<td>Explicit CF ownership, dividing into distributed float (DF) according to square root of (dur · CI)</td>
</tr>
<tr>
<td></td>
<td>Activities on subcritical paths already have float</td>
</tr>
</tbody>
</table>
VALIDATION
Simulations for a 15 activity schedule (Thompson and Lucko 2011) and a randomly selected more complex J60 from the PSPLIB are run 100 runs for CCPM (with fixed baseline) and the new approach (with varied CF) per Table 1. For all activities a triangular distribution with 90% and 150% of original durations is applied. Figures 1 and 2 show the PB performance of the 15 activity example and J60. Most runs exceed PB, which here is divided into traffic light colored thirds per Dilmaghani (2008). A further analysis shows how side paths cause it by exceeding FB (those figures are omitted for brevity). Overall, the static CCPM deploys buffers inefficiently in a single fixed ‘first come first serve’ offer to the project. It performs poorly in both schedules.
Figures 3 and 4 are dynamic views of the new allocation, showing how much critical activities exceed their DF (y-axis) when varying CF (x-axis). They compare a square root approach (exponent 0.5) with equal float allocation (exponent zero) and proportional float allocation (exponent one). Since different activities are ‘saturated’ at different CF, the overall performance of the bundles of decreasing curves must be compared. The minimum and maximum CF where the decrease starts and finishes are extracted. Then the mean and standard deviation for these intervals are calculated.

The square root approach appears to be effective, because it has a lower mean and standard deviation than either the exponent zero or one. It also shows that CCPM takes a too narrow view. Using a product duration and CI to reflect interconnectivity via the behavior of the project schedule improves the approach over an earlier attempt that has been reported by Lucko and Thompson (2015). It holds true both for the 15 activity example (9.07 and 12.22 for exponent 0.5; 9.6 and 13.47 for exponent 0; 9.9 and 13.87 for exponent 1) and the more complex J60 (5.43 and 11.17; 6.33 and 13.71; 5.98 and 12.48). The agreement of these two results at different complexities implies a more general support. Still, more simulations are needed. Further examination of how the square root of duration and CI will perform under more varied conditions is planned. These simulations to validate the new approach fulfill Research Objective 3.

CONTRIBUTIONS
Existing approaches reveal a continued need to examine buffer location and sizing strategies and float ownership allocation. New inspiration has been gleaned from the voting power concept, which seeks a balance between extremes via the square root. Contributions have included that CCPM has been criticized, tested, and found to be too static, simplistic, and narrow. Therefore, a more dynamic approach that seeks to balance between the extremes of ‘equal’ and ‘proportional’ float allocation for risk mitigation has been introduced, simulated, and found to provide promising results.

Future research should: 1. Run more simulations across the spectrum of various complexity metrics to explore how network schedules perform. 2. The new approach should be broadened to a generalization of its square root: Since activities reside in a mixture of sequential and parallel positions in a schedule, it should be investigated how they benefit or suffer. In other words, delays from early activities will percolate downstream to later ones; the last one is arguably in an unfavorable position, where it ‘inherits’ their cumulative impacts. It is envisioned for the exponent 0.5 of the square root to be adjustable, so that earlier activities get slightly less and later ones slightly more DF allocated. To what extent and in what (non)linear shape such ‘unbalanced allocation’ would perform best requires more simulations. Inspiration may be gained from unbalancing profit margins in cash flow research. 3. Instead of assuming CF as a fixed value per the contract, it should be asked what amount of CF would optimally protect a schedule. This requires varying CF incrementally in a simulation, testing the schedule resistance, and – inspired by PERT – allowing CF for a level of confidence that the user finds acceptable. 4. Assuming independent activities should be replaced with conditional dependencies between predecessor-successor pairs, which would require richer inputs, if users can provide them, but could enhance the realism of the analysis. 5. The new approach should examine how feeder paths can be protected.
ACKNOWLEDGEMENT
The support of the National Science Foundation (Grant CMMI-1265989) for portions of the work presented here is gratefully acknowledged. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES


