

Roshan POUDEL, Borja GARCIA de SOTO, Eder MARTINEZ

Last Planner System and Scrum: Comparative analysis and suggestions for adjustments

© Higher Education Press 2020

Abstract This study provides a critical review of the concepts of Agile, Lean, Scrum, and Last Planner® System (LPS). A comparative analysis is conducted between LPS and Scrum to expand LPS by considering Scrum's best practices. Eight dimensions, namely, 1) origins, 2) main purpose, 3) overall system/framework process, 4) tools or artifacts maintained by the team, 5) team composition and main roles, 6) regular events or team meetings, 7) metrics/dashboards, and 8) approach to learning, are evaluated. After analyzing side by side the eight dimensions, it was found that many aspects from Scrum already exist in LPS in the same or similar form. However, the authors identify four main elements from Scrum that can be leveraged to improve the LPS benchmark, such as considering the Scrum "Increment" concept into LPS, having a clear definition of roles and responsibilities, or adding an equivalent to a Scrum Master to have a designated "rule keeper" in LPS. These opportunities to be considered in new LPS benchmarks need to be tested and validated with real applications. To the best of the authors' knowledge, this work is the first to comprehensively compare Scrum (Agile) and LPS (Lean) and could be seen as a contribution toward the evolution of the Last Planner System for the academic and industrial environments.

Keywords Lean Construction, Last Planner System, Agile, Scrum, comparative analysis, AEC projects, project teams

Received March 8, 2020; accepted April 16, 2020

Roshan POUDEL, Borja GARCIA de SOTO (✉)
Division of Engineering, S.M.A.R.T. Construction Research Group,
New York University Abu Dhabi (NYUAD), Saadiyat Island, Abu
Dhabi, United Arab Emirates
E-mail: garcia.de.soto@nyu.edu

Eder MARTINEZ
Lean Construction Specialist, Implenia Group, Industriestrasse 24, CH-
8305 Dietlikon, Switzerland

1 Introduction

Traditionally, the architecture, engineering, and construction (AEC) industry uses waterfall contractual models and methods for project management. Contractual models, such as design-bid-build, design-build, construction management at risk, or a combination of them, are common means to define the stakeholders' relationships (Sanvido and Konchar, 1999) along the project lifecycle (e.g., design, construction, and handover to the owner). In these models, the owner generally hires an architect (or a contractor with design capabilities) to define the building design and specifications. As the design progresses, additional stakeholders are brought into the project based on discrete bidding processes for different trades and specialties. Traditional project management methods, such as work breakdown structure, critical path method, precedence diagramming method (PDM), and earned value analysis, are also used indistinctly to plan and track a project's progress. Although PDM is perhaps the most common way to conduct project scheduling (Hajdu, 2018), it presents some limitations, such as the assumption of linearity and the logic connections used, usually at the endpoints of tasks (Hajdu et al., 2017). In all cases, these methods require thorough upfront planning and a certain degree of predictability to be effective. Ideally, the project scope, targets, budget, and risks should be soundly defined at the project conception to avoid deviations. However, such a level of predictability is rarely achieved in construction projects (García de Soto et al., 2017).

Arguably, these types of contracts and management methods are well suited to fit the linear nature of AEC projects. However, the transition of the AEC industry toward more integrated forms of project delivery and technological advances in the modern built environment challenges the traditional way of planning and managing projects. Facility owners, designers, contractors, and subcontractors are increasingly realizing that collaboration from early design through project handover is the cornerstone to deliver high-performing facilities (AIA,

2007; Fischer et al., 2017). Furthermore, the evolution of the industry toward Construction 4.0 (García de Soto et al., 2019), smart buildings, Building Information Modeling (BIM) (Sacks et al., 2010), and the Internet of Things adds another layer of complexity to the project delivery process because it requires interfaces with product, software, and service development. This situation also raises the issues of cybersecurity (Fisk, 2012; Boyes, 2013; Mantha and García de Soto, 2019; Parn and Edwards, 2019). Thus, the traditional methods used to plan and control the delivery of modern infrastructure should be updated. Teams need to use the best collaboration systems and tools to ensure that the facility conforms to customer requirements while meeting project cost, time, and sustainability targets.

Due to the widespread adoption of BIM, many researches (Sacks et al., 2010; Hamdi and Leite, 2012; Oskouie et al., 2012; Dave et al., 2013) have investigated the synergies between Lean and BIM and developed applications to integrate them and improve the productivity of construction projects by applying both approaches concurrently. Examples of tools developed include the Lean Enterprise Web-based Information System (LEWIS) (Sriprasert and Dawood, 2003), the VisiLean tool (Dave et al., 2011), and the KanBIM Workflow Management System (Sacks et al., 2013). Heigermoser et al. (2019) developed a prototype tool to integrate the Last Planner[®] System (LPS)¹⁾ workflow using BIM and provide planning visualization and evaluation features, which are particularly related to the short-term planning and learning phases in the LPS.

In the early 1990s, LPS emerged under the umbrella of Lean Construction as an alternative to traditional planning and production control systems (Ballard, 2000). Since then, several case studies have been conducted and have provided valuable inputs to improve the original LPS framework. As part of the evolution of LPS, understanding how LPS can be adapted to support teams to collaborate within a more dynamic environment for delivering modern infrastructure is important. In the latest LPS process benchmark, Ballard and Tommelein (2016) proposed analyzing Scrum to explore what elements of this Agile framework can be used to improve LPS. Scrum has been primarily used in the information systems and manufacturing industry; however, except for a handful of exceptions (Lia et al., 2014; Demir and Theis, 2016; Kalsaas et al., 2016; Streule et al., 2016; Ormeño Zender and García de Soto, 2020), the potential of Scrum in the AEC industry has not yet been fully explored.

The current study builds upon Ballard and Tommelein (2016)'s request for research and contributes to the existing literature by conducting an exhaustive analysis and comparison of LPS and Scrum. Through the literature review, the authors analyze different dimensions of LPS

and Scrum to highlight their similarities and differences. In this way, elements from Scrum that can be potentially used to improve LPS can be found. The paper is structured as follows. The authors analyze and describe the main principles of Lean-LPS and Agile-Scrum, followed by a review of the literature that reports their use in the AEC industry. Thereafter, LPS and Scrum are compared side by side on the basis of eight different dimensions (e.g., team composition, roles and responsibilities, meetings/events, metrics). On the basis of this comparison, the authors discuss similarities and differences to discover Scrum elements that can be used to improve LPS. The authors provide insights for future research on the basis of the areas for improvement.

1.1 Lean and Agile overview

Lean refers to a theory that aims to increase the value for the customer while conserving resources and minimizing waste. The principles of Lean emerged after World War II when Taiichi Ohno led Toyota to build a great quantity and variety of cars with less labor, capital, and inventory (Krafcik, 1988). This mechanism became known as the Toyota Production System (Womack et al., 1990). Although Lean theory initially focused on the automobile manufacturing industry, it has since been applied in numerous other industries. In the early 1990s, Koskela (1992; 2000) argued that several Lean production principles and methods could be adopted in the construction industry. He proposed a production theory for construction by combining three complementary views, namely, transformation, flow, and value, and argued that all should be considered simultaneously. Under the transformation view, production systems are broken down into activities, and performance is gauged on the basis of how well each activity is performed independent of the others. Under the flow view, production systems are perceived as a series of interconnected processes in which raw materials gradually move from one process to another to become finished products for the customer. Performance also considers this interconnectedness of activities and buffers. Under the value view, the objective is to achieve the maximum possible value for the customer. The customer is thus integrated into the production system. Koskela's work set the theoretical foundations of Lean Construction (Ballard and Howell, 2003a).

Agile refers to a set of iterative, incremental, and value-oriented development methods that are rooted in the *Manifesto for Agile Software Development* (Beck et al., 2001). On the basis of the "principles", Agile methodologies seek early delivery of value through frequent iterations supported by cross-functional and self-organized teams. Agile emerges in the information systems industry as a response to the many drawbacks

1) Last Planner[®] is a registered trademark of Lean Construction Institute.

experienced by practitioners in applying traditional project management methods (e.g., waterfall) to software development (Abbas et al., 2008). Traditional approaches seek to anticipate and freeze requirements for managing the project time, cost, and quality, while Agile welcomes change and uses it as an opportunity to enhance the value delivered to the customer (Highsmith and Cockburn, 2001). Several concepts embedded in Agile date back to the 1970s (Royce, 1970). However, the formalization of the Manifesto by leaders in the information systems industry triggered the spread of Agile Project Management ideas, even beyond software development domains (Sanchez and Nagi, 2001; Ciric et al., 2018).

1.2 Lean and Agile in the construction industry

When compared to Lean, Agile has not been significantly investigated in the construction industry. Owen and Koskela explored the applicability of Agile in construction and identified potential benefits in its use during pre-design and design phases of AEC projects (Owen and Koskela, 2006; Owen et al., 2006). Barlow (1998) explored the use of an Agile system during the conceptualization phase of a housebuilding project. He found that the Agile system can react faster to changing customer requirements than the traditional approaches for project management. In a similar context, Naim et al. (1999) explored the Agile and Lean systems by using the “leagility” concept in the UK housebuilding industry. Lessing et al. (2005) also reported a case study in the housebuilding industry. The authors found that this industry is well suited and better than other construction sub-sectors for Agile implementation. Other studies have explored the use of Agile in offsite construction (Mostafa et al., 2016) and in combination with BIM (Lu et al., 2011).

Under the Lean Construction umbrella, LPS is the most commonly used method in the AEC industry. Several case studies have reported the positive effects on planning reliability, workflow predictability, and production performance (Tommelein and Ballard, 1997; Ballard and Howell, 1998; Ballard et al., 2009). In the Agile arena, the most commonly used framework is Scrum. Scrum has been widely implemented in software and hardware development but not yet fully explored in the construction industry. Only a few studies have reported Scrum implementation in the AEC industry, and most of them do not consider leveraging potential synergies with LPS.

The next sections build upon previous research on the LPS and Scrum in the AEC context to analyze and compare them side by side. The ultimate purpose is to find opportunities for improvement to the LPS benchmark and thus address the need for the research proposed by Ballard and Tommelein (2016). Finding synergies between LPS

and Scrum has the potential to derive a project management approach capable of supporting teams in collaborating to deliver modern infrastructure.

2 Research methodology

The methodology for this research is based on a thorough literature review. The authors searched relevant publications related to the principles and applications of Agile, Lean, LPS, and Scrum. The primary source for the literature review was the core collection of Web of Science (WoS) from 1990 to 2019. The *Lean Construction Journal*¹⁾ (*LCJ*) and the *proceedings of the International Group for Lean Construction*²⁾ (*IGLC*) are not part of the WoS database. Thus, the authors also went through the repositories for *LCJ* and *IGLC* to obtain relevant publications.

The terms used to identify relevant publications in WoS were as follows: LPS, Last Planner System, Lean Project Management, Agile Project Management, Scrum, Similarities/Comparison between Lean and Agile, Comparison between LPS and Scrum, History of Lean, History of Agile, Use of Agile in Construction Industry, and Use of Lean in Construction Industry. After the search, articles were analyzed to elaborate on the description of the relevant concepts serving as the foundation to compare LPS and Scrum. In the case of LPS, the search was filtered using the following categories: Engineering Civil, Construction Building Technology, Engineering Industrial, and Engineering Manufacturing. Similarly, for Scrum, the search was filtered using the following categories: Computer Science Software Engineering, Computer Science Theory Methods, Engineering Electrical Electronics, Computer Science Information Systems, Computer Science Interdisciplinary Applications, Computer Science Artificial Intelligence, and Management. The publications on Lean were refined using the categories of Engineering Industrial, Engineering Multidisciplinary, Engineering Manufacturing, Engineering Mechanical, Engineering Civil, and Construction Building Technology. The refinement of categories for Agile was the same as that for Scrum. Since there were not many publications related to the comparison between Agile and Lean, no further refinement was made for those. The overall process is summarized in Fig. 1.

The literature review revealed the volume of publications for the different topics considering all the sources used (i.e., WoS, *LCJ*, and *IGLC*). As expected, the number of publications related to Lean and Agile topics in the categories specified is significant. From 2015 to 2019, the average number of publications per year was 762 for Lean and 918 for Agile. Manuscripts in the field of Scrum were

1) Publications available online from 2004 to 2019 (2006 did not have any publication).

2) Publications available online from 1996 to 2019.

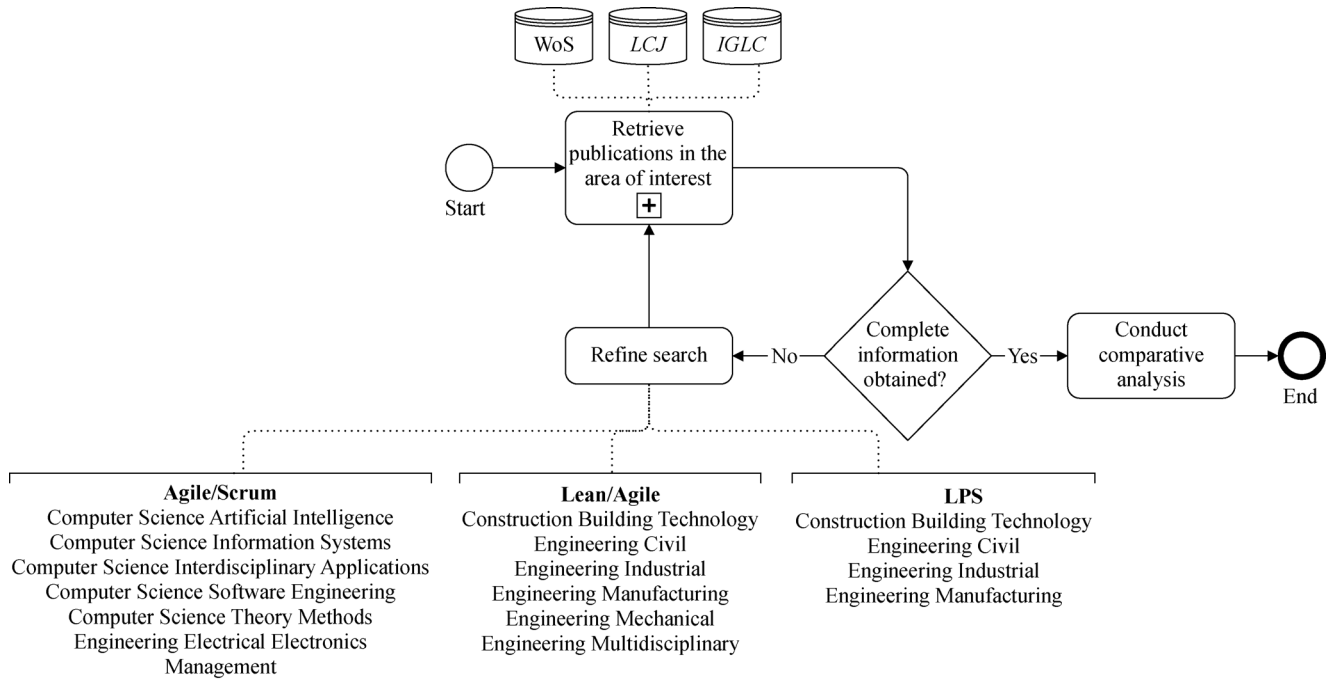


Fig. 1 Flowchart for the literature review process.

higher than those for LPS, with an average of 125 and 21 publications per year, respectively, from 2015 to 2019. The number of publications focusing on the comparison of Agile and Lean was very limited, with an average of 4 publications per year from 2015 to 2019. The authors could not find any publication comparing LPS and Scrum. Figure 2 shows the number of publications versus the

publication year for the topics of Last Planner System (LPS), Scrum, Lean, Agile, and Agile vs. Lean from 1990 to 2019. The right y-axis corresponds to the number of publications in the topics of LPS and Agile vs. Lean. These manuscripts have been separated for clarity due to the relatively lower number of publications when compared with the other topics.

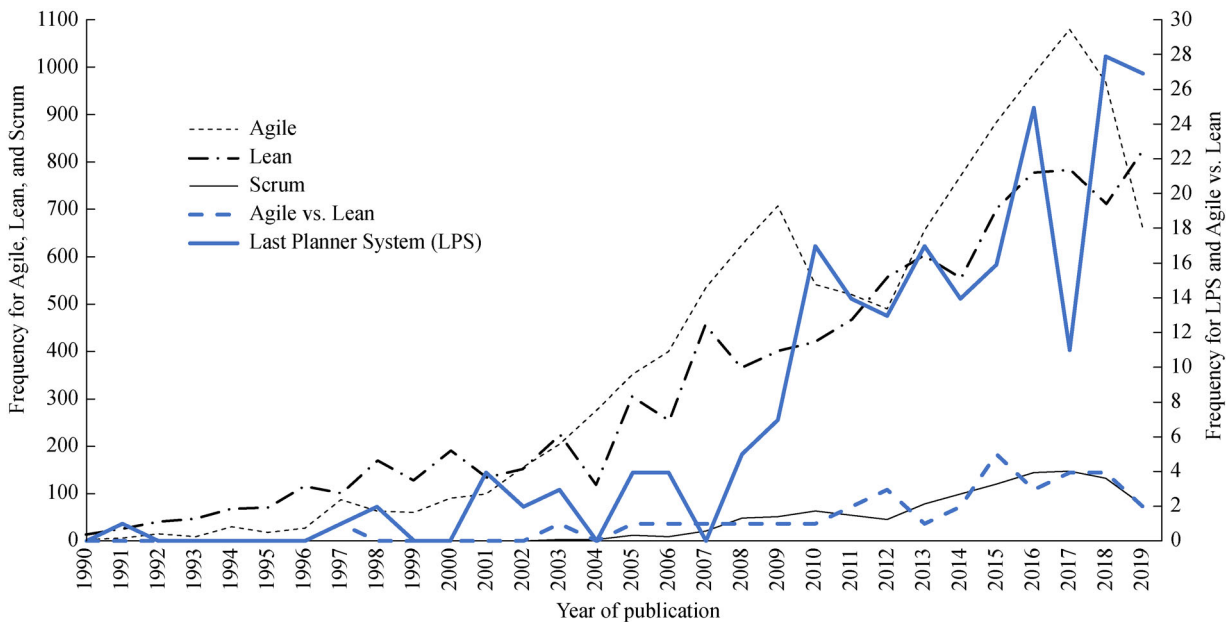


Fig. 2 Number of publications for manuscripts related to the research areas of interest. The left y-axis shows the number of publications for Agile, Lean, and Scrum (thin lines). The right y-axis shows the number of publications for LPS and comparative studies between Agile and Lean (thick lines).

To facilitate the comparative analysis, the authors defined eight different dimensions: 1) origins, 2) main purpose, 3) overall system/framework process, 4) tools or artifacts maintained by the team, 5) team composition and main roles, 6) regular events or team meetings, 7) metrics/dashboards, and 8) approach to learning.

2.1 Last Planner System

LPS is a comprehensive and integrated system for planning and production control. The first draft of LPS was introduced by Ballard and Howell in the early 1990s, and it was tailored to the AEC industry based on Lean production principles (Ballard et al., 1996; Ballard and Howell, 1998; Ballard, 2000).

The different components, functions, and related processes of the LPS are summarized in Fig. 3. The system consists of five levels of planning. In the master schedule at the top level, the team defines the main milestones early in the project. These milestones set the pace of progress according to the defined project targets. In phase scheduling, the team uses pull planning to further detail the sequencing of the main portions of the project, and the primary project stakeholders agree on the conditions of satisfaction for different handoffs among milestones. Pull planning is a method to collaboratively define the project plan involving those responsible for performing the work (as opposed to one person estimating the project duration). This part includes planning backward from the target completion date milestone by milestone and defining tasks, sequence, and condition of satisfaction for

work release. In the master and phase scheduling levels of planning, the team formulates what “should” be done. Both planning levels are updated as needed on the basis of the project progress.

Lookahead planning focuses on sequencing and sizing the work that “can” be done. For this purpose, the team develops a lookahead plan, which is usually in a 6-week time window (depending on project complexity). This plan is used to identify and remove activity constraints. In the lookahead plan, the team also refines task descriptions and breaks them down into manageable pieces to facilitate the design of operations. In commitment planning, the team collaboratively develops the Weekly Work Plan (WWP) by committing to the tasks that “will” be done during a given week. Reliable promising is a key concept in LPS, where the task requester and performer must be clear about what has to be done and delivered. The last level of planning focuses on learning from what the team “did”. This part is performed by analyzing the main reasons of plan failures and defining countermeasures accordingly.

The only fixed role in LPS is the one assigned to “last planners”. In general, the last planners are experienced team members who have knowledge about how to optimally perform the work being planned. This group includes different specialists, frontline supervisors, and craftworkers. Everybody that can contribute to improving reliability in planning and production design can play a “last planner” role. This group includes team members with different types of expertise, such as material or equipment sourcing, safety, quality, and logistics. The “last planner” role is played by different team members

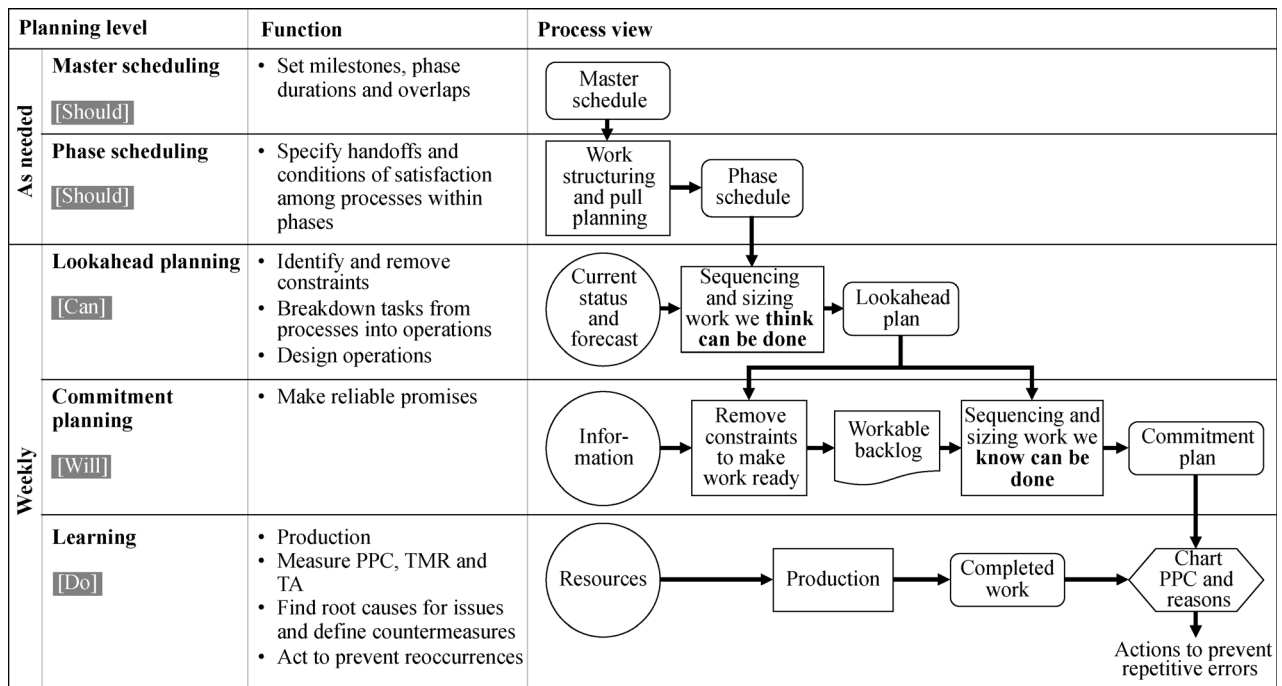


Fig. 3 Overview of the LPS planning levels and related process (adapted with permission from Ballard (2000)).

depending on the type and progress of the project. For example, during the design phase, the last planners are typically architectural and engineering specialists. During the construction phase, the last planners are typically foremen and trade superintendents or supervisors (Ballard and Tommelein, 2016).

LPS has four main metrics to gauge planning system performance: Percent Plan Complete (PPC), Tasks Made Ready (TMR), Tasks Anticipated (TA), and Frequency of plan failures. PPC measures workflow reliability, which is computed by contrasting the tasks done against those that were planned in the WWP. TMR measures the team's capacity to identify and remove constraints in the lookahead plan and thus make tasks ready for execution. Its computing is similar to that of PPC but is only done earlier by comparing the WWP against an earlier week in the lookahead planning. TA measures the team's ability to anticipate tasks that will be committed, such as 4 or 5 weeks in the future (Hamzeh and Aridi, 2013; Ballard and Tommelein, 2016). The Frequency of plan failures metric seeks to track the causes for tasks not done in the WWP. The reasons of plan failures are clustered into categories (e.g., weather, design, equipment, materials). As plan failures occur, the team can compute their relative frequency, which provides insights about the potential root cause for problems. For instance, a high frequency in the equipment category provides the team insights to reveal root causes and define countermeasures (Ballard and Tommelein, 2016).

Applications of LPS

Several case studies have assessed the outcomes of LPS implementation in different project contexts. Considerable literature is available in *the proceedings of the IGLC*. Predominantly, articles highlight positive correlations between improved workflow reliability resulting from LPS implementation and an increase in production productivity (Gonzalez et al., 2008; Liu and Ballard, 2008; Liu et al., 2011). A cluster of case studies has provided evidence of positive correlations on safety, quality, and cost performance. However, additional data are needed to support these correlations (Ballard and Tommelein, 2016).

Some researchers have also reported on LPS challenges and drawbacks. El Samad et al. (2017) argued that most projects use only the PPC to assess LPS performance. This approach results in a misleading overview of the overall project planning and production performance. When entering the WWP, the PPC metric assumes that all activities are equally relevant to project progress, which is not always the case in practice. Dave et al. (2015) highlighted several LPS implementation issues related to partial LPS implementation, such as only using the commitment planning level, disconnection among different planning levels, lack of collaborative involvement of "last planners", ambiguity in planning roles and respon-

sibilities, and difficulties in tracking and monitoring metrics. The authors concluded that the full potential of the LPS is rarely achieved because of partial implementation. Deficient LPS implementations may naturally affect some of the positive correlations reported on LPS.

LPS has also been explored during the design phase of projects. Ballard et al. (2009) acknowledged the differences between designing and construction phases and recommended that some LPS adaptations are required to cope with the increased uncertainty, speed, and complexity inherent to the iterative design process. Case studies about LPS used during design have reported increased planning reliability and reduced workflow variability. LPS implementation in design remains behind the level of maturity it has achieved in the construction phase; thus, further research is required to close the gap (Koskela et al., 1997; Ballard, 2000; Hamzeh et al., 2009; Tiwari and Sarathy, 2012; Khan and Tzortzopoulos, 2015).

Lean Construction researchers and practitioners have regularly collaborated to strengthen LPS theoretical foundations and support its implementation in real-life contexts. They have investigated and developed LPS benchmarks and consolidated the main findings, challenges, and opportunities for improvement to foster LPS implementation and enlighten future research (Ballard and Tommelein, 2016).

2.2 Scrum

Scrum is an Agile process framework used to manage complex product development projects where unpredictability is exacerbated due to uncertainty in requirements and technology. The ultimate purpose of Scrum is to support project teams on delivering products with the highest possible value. Scrum is founded on empirical process control theory using an iterative and incremental approach to optimize predictability and manage project risks (Schwaber and Beedle, 2001; Cervone, 2011). The term Scrum appeared for the first time in a paper by Takeuchi and Nonaka (1986). In 1995, the Scrum framework was formalized by Sutherland and Schwaber in an information systems conference (Schwaber and Beedle, 2001).

Figure 4 depicts the Scrum framework. The framework consists of Scrum roles, Scrum artifacts, and Scrum events. The three Scrum roles are the Product Owner, the Development Team, and the Scrum Master. The Product Owner is typically a single person responsible for maximizing the value of the product and the work of the Development Team. The Development Team is a cross-functional, self-organizing team responsible for doing the actual work to develop the product. The Scrum Master ensures that everyone in the Development Team understands the theory, practices, and rules of Scrum. The Scrum Master is also responsible for removing obstacles that may affect the progress of the Development Team (Schwaber

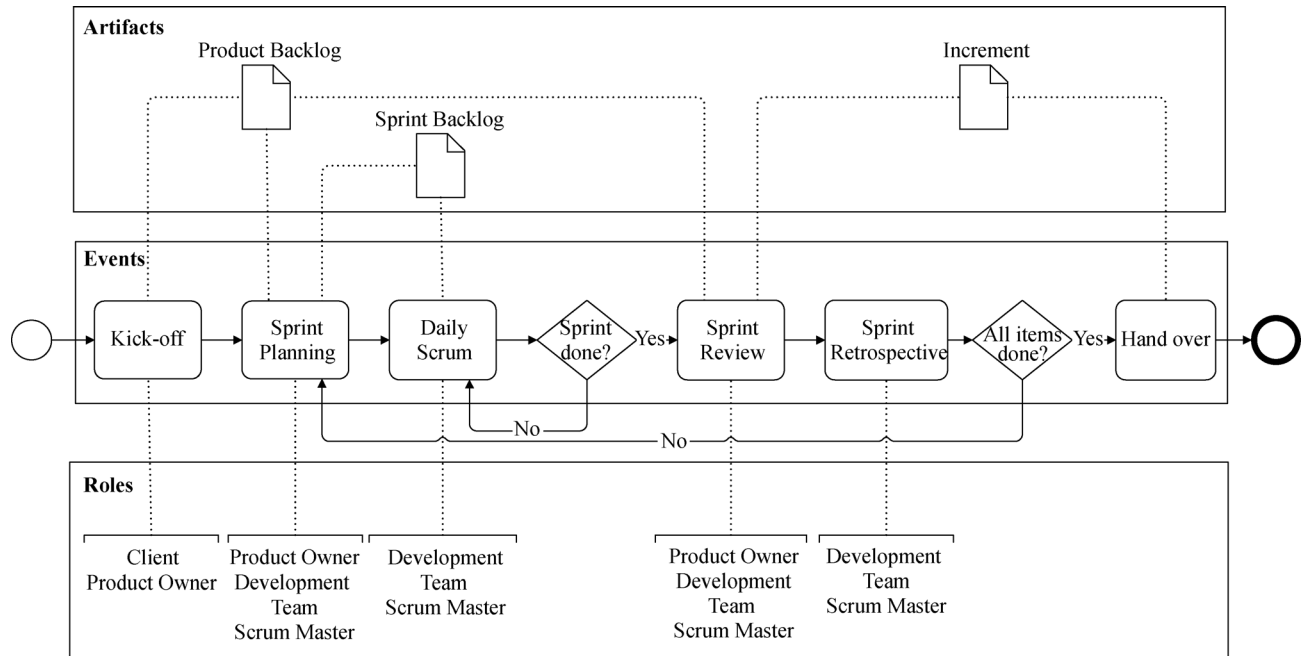


Fig. 4 Scrum process and related elements (adapted with permission from Streule et al. (2016)).

and Sutherland, 2013).

Scrum artifacts are the Product Backlog and the Sprint Backlog. The Product Backlog, owned by the Product Owner, is a prioritized list of everything that may be needed in the product. The Sprint Backlog is the set of the Product Backlog items that have been selected by the Development Team for a given Sprint, along with a plan to achieve the Sprint Goal (Rubin, 2012; Schwaber and Sutherland, 2013).

Scrum events are the Sprint, the Sprint Planning meeting, the Mutual Commitment, the Sprint Review, and the Sprint Retrospective meeting. The Sprint is a time window of one month or less in which the team sets the target to deliver a product increment. The Scrum team plans for the work to be done in the Sprint Planning meeting on the basis of the defined product increment. The Mutual Commitment is a commitment made by the Product Owner not to change the Sprint Goal and by the Development Team to complete the Sprint Goal by the deadline set by the team (Rubin, 2012). During the Sprint, the team performs Daily Scrum meetings. In this short 15-min event, the team members discuss the work that has been done since the last Daily Scrum and the work that will be done until the next Daily Scrum. They share information to keep everybody on the same page and identify potential road blockers. At the end of a defined Sprint, the team reviews the tasks “done” along with the remaining Product Backlog items, and they revise and update the Product Backlog if required. It is important to note that the Development Team defines what is required for something to be labeled “done”, and all the members must understand and agree upon the definition of “done”.

After the Sprint is completed, the team gets together in the Sprint Retrospective meeting to learn from the Sprint experience; thus, learnings and opportunities for improvements can be embraced and implemented in the next Sprint (Schwaber and Sutherland, 2013).

Scrum includes several metrics to track project progress and performance. The most important one is team velocity. Velocity is a metric that is used to track the amount of Product Backlog effort that a team completes in a Sprint. The team gauges the effort required in terms of story points. Story points represent a combination of the amount of effort involved in developing the feature, the complexity of developing it, the risk inherent in it, and, most important, the value delivered to the customer. This is an alternative to the traditional measure effort in terms of time related to tasks. In story points, the raw value assigned to tasks is not as relevant as the relative value among them (Cohn, 2005). For example, a task that doubles another in terms of complexity, risk, and value delivered should be assigned double story points when compared with its counterpart (e.g., 1 vs. 2, or 7 vs. 14, or 5 vs. 10). Planning poker (Grenning, 2002) is a popular and reliable method that Scrum teams use to collaboratively derive reliable story point estimates. Scrum teams favor using story points due to technological uncertainty related to projects. The functionality needed for the product may be known, but the level of effort required to deliver it may vary according to the realization of the technology that happens within the same project. The value to the project is then easier to measure than the resources needed to deliver the required functionality.

Several metrics and visualization methods are derived

from the story points and commitments made by the team. The Task Board is used to show the Sprint execution progress by categorizing tasks as “to do”, “in progress”, and “completed”. The Sprint Burndown Chart is a visual representation of the amount of effort that is still required to complete the Sprint (Rubin, 2012). All these metrics and visualization methods are consolidated and shared through the “Information Radiator”.

Applications of Scrum

Scrum is widely used for software development projects. In this context, Cardozo et al. (2010) reviewed existing literature to search for scientific evidence about outcomes after the implementation of Scrum. From 274 articles, the authors selected 28 for further analysis, which offered strong evidence that Scrum may have a positive correlation with software development productivity according to their research methodology. The authors reported “incidental benefits” in customer satisfaction, product and process quality, team motivation, and cost reduction. Similarly, Caballero et al. (2011) analyzed the Scrum productivity effects on a small software development enterprise. After the man-hours spent in software development, cycle times, and software code quality were analyzed, the authors reported productivity gains without affecting the quality of the final software product.

Scrum is also used in manufacturing. “Extreme manufacturing”, a term coined by Joe Justice and his Wikispeed team, uses a combination of Scrum and Lean principles to build car prototypes. Extreme manufacturing gained popularity after the Wikispeed team at the Progressive Insurance X Prize competition in 2010 and at the Detroit Auto Show in 2011 (Kupp et al., 2015). This approach has been successfully used by multinational companies, such as Boeing and General Motors (Kupp et al., 2015).

Along with the growing interest in Scrum application, several drawbacks have also been identified. Most of the successful cases are based on small co-located teams. However, research has revealed several challenges when implementing Scrum at a large scale. Hossain et al. (2009) emphasized that decentralized global software teams struggle to cope with the frequency of team meetings involved in the Scrum framework. As a result, the team must look for workarounds to synchronize people working in different time zones. This situation affects rapid team feedback, which is fundamental in the Scrum framework and demands additional project management tools to track the progress of work for the decentralized team members. In other cases, teams tend to focus on following the process and “checking the boxes” instead of concentrating on booster creativity in delivering value to the customer (Cohn, 2014).

According to the authors’ experience in implementing Scrum in global manufacturing firms, it is very common to see teams forcing the Scrum framework to fit into rigid

organizational structures. Large firms exploring Scrum have neither Scrum roles nor adequate decision-making schemes to facilitate Scrum implementation. This derives in several “hybrid” Scrum approaches that deviate from the original framework and rules. These hybrid Scrum approaches are even observed in large software firms (Iamandi et al., 2015). Although the authors acknowledge that this situation may also derive from an inadequate Scrum introduction approach, the ultimate consequence is that documented case studies are difficult to compare. Thus, the quality of the data for deriving conclusions is affected.

Only a few case studies have documented the use of Scrum in the construction industry. Lia et al. (2014) combined ideas from LPS, Scrum, and Critical Chain to increase predictability in the delivery of complex engineering projects. Kalsaas et al. (2016) proposed Scrum to establish short milestones and iterations to proactively address changes occurring during the design phase. Demir and Theis (2016) proposed a combination of Scrum in an Agile design model to increase coordination, interface management, collaboration, and transparency during the project design. The case study by Streule et al. (2016) investigated the implementation of Scrum during the project definition, profitability optimization, and cost optimization phases. The authors reported that the project team valued Scrum because it enabled them to learn more about topics beyond their expertise and understand the reasons for various actions taken in the project. Some disadvantages addressed by the project participants include the lack of a distinct project leader and the delays due to the voting issues when making decisions. Ormeño Zender and García de Soto (2020) applied Scrum in the rehabilitation of a commercial building. Due to the high uncertainty related to the project and the lack of information available, it was an ideal scenario to implement an Agile method. Their study showed that Scrum provided flexibility to address changes (induced by the client or by the uncertainty of the context in which the project was developed) during the rehabilitation work. They also found that Scrum provided good risk control, allowed the project team to complete all the rehabilitation work within a tight schedule, and improved the general satisfaction of the stakeholders involved. However, the overall Scrum implementation in the AEC industry is far behind in terms of maturity compared with software and hardware development. Most studies to date are exploratory. Except for Lia et al. (2014), Scrum is evaluated in isolation, and potential synergies with a well-recognized system in the AEC as LPS are not explored or leveraged.

2.3 Comparison between LPS and Scrum

On the basis of the analysis of the LPS and Scrum literature, the authors compare them in terms of eight different dimensions, namely, 1) origins, 2) main purpose,

3) overall system/framework process, 4) tools or artifacts maintained by the team, 5) team composition and main roles, 6) regular events or team meetings, 7) metrics/dashboards, and 8) approach to learning. A summary of the comparison is provided in Table 1 and is further expanded in this section. This comparison is the baseline to discuss potential Scrum elements that can be used to improve LPS in Section 3.

(1) Origins: LPS was developed for the AEC industry, and it is rooted in Lean Production principles. LPS applications started in the construction phases but incrementally evolved toward design phases of construction projects. Early Scrum concepts emerged from the manufacturing industry (Takeuchi and Nonaka, 1986). Later, the Scrum framework was refined and tailored to the software industry. After the popularity of Scrum increased, it found its way back to the manufacturing industry.

(2) Main purpose: LPS and Scrum aim to increase the value delivered to the customer.

(3) Overall system/framework process: LPS and Scrum use different planning levels to refine and coordinate the activities necessary to deliver the project/product. Activities are defined at a high level, sized, and further detailed as the team screens them through the different planning levels by using the corresponding tools and artifacts. Both methods rely on team members' inputs to collaboratively define the completion of activities on the basis of the reliable promising/commitment concepts. LPS and Scrum also emphasize systematic learning to improve team performance.

(4) Tools or artifacts maintained by the team: In the LPS, the master and phase schedule represent what the team "should" deliver, that is, the milestones and phases necessary to realize the end product (Ballard and Howell,

Table 1 Summary of the comparison between LPS and Scrum

Dimension	LPS	Scrum
1. Origins	<ul style="list-style-type: none"> • In the construction industry • Based on Lean production principles 	<ul style="list-style-type: none"> • In the manufacturing and software industry • Aligned to the <i>Manifesto for Agile Software Development</i>
2. Main purpose	<ul style="list-style-type: none"> • Increase value for the customer while reducing waste 	<ul style="list-style-type: none"> • Deliver products of the highest possible value for the customer
3. Overall system/framework process	<ul style="list-style-type: none"> • All activities necessary to complete the project are broken down and refined through the different planning levels • Proactive identification and removal of activity constraints • Sequencing and sizing of work based on reliable promising to improve workflow derive in the WWP • Analysis of plan failures for continuous improvement 	<ul style="list-style-type: none"> • Everything needed to deliver the product (requirements) is managed, refined, and prioritized in the Product Backlog • Team commits to work on specific Product Backlog items deriving the Sprint plan • The team works to deliver product increments iteratively • Analysis of Sprint experiences for continuous improvement
4. Tools or artifacts maintained by the team	<ul style="list-style-type: none"> • Master schedule • Phase schedule • Lookahead plan • Workable backlog • WWP 	<ul style="list-style-type: none"> • Release plan • Product Backlog • Sprint Backlog • Increment
5. Team composition and main roles	<ul style="list-style-type: none"> • "Last planners" who are knowledgeable team members providing inputs about how to optimally perform the work (e.g., construction specialists, frontline supervisors, craftsmen) • Number not specified 	<ul style="list-style-type: none"> • Product Owner • Development Team • Scrum Master • Recommendation of less than nine team members
6. Regular events or team meetings	<ul style="list-style-type: none"> • Planning meetings (cadence not strictly defined for different planning levels) • WWP (planning and review) • Daily huddles 	<ul style="list-style-type: none"> • Envisioning (Product Planning) • Scrum Release Planning • Sprint Planning (cadence according to Sprint duration for every Sprint) • Sprint Review • Sprint Retrospective • Daily Scrum
7. Metrics/Dashboards	<ul style="list-style-type: none"> • PPC • TMR • TA • Frequency of plan failures • Visual controls to consolidate and share project information 	<ul style="list-style-type: none"> • Velocity • Task Board • Sprint Burnup Chart • Sprint Burndown Chart • Information Radiators to consolidate and share project information
8. Approach to learning	<ul style="list-style-type: none"> • Analysis of Frequency of plan failures • 5 Whys, Plan–Do–Check–Act, Detect–Correct–Analyze–Prevent 	<ul style="list-style-type: none"> • Sprint Retrospective

2003b). In Scrum terminology, this “should” corresponds to everything that is necessary to deliver the product (in terms of requirements). Therefore, the master and phase schedules in the LPS serve a similar purpose as the Scrum Product Backlog.

The proactive identification and removal of activity constraint function in the lookahead plan in the LPS are also fulfilled in the Scrum Product Backlog. The Scrum Product Backlog contains activities with different levels of detail that the Scrum team continuously refines until they can reasonably be “done” within the Sprint time-box. This corresponds to the workable backlog of activities that are ready to be discussed by the team and incorporated in the WWP and Sprint Backlog. Therefore, the LPS WWP is fulfilled in the Scrum Sprint Backlog because both items contain the work the team plans to complete during a week/Sprint.

The Scrum Increment concept is not covered in the LPS. An Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints (Schwaber and Sutherland, 2013). Arguably, in LPS, an Increment may be a useful concept during the design phase, where specialists can define the completion of the design of a given building area in the course of a week. However, the concept is less applicable at the construction phase because the majority of construction work (e.g., a reinforced concrete wall or room) takes longer than a week and involves several trades. Thus, an Increment may be unachievable in a week (or Sprint).

(5) Team composition and main roles: LPS and Scrum emphasize team collaboration in the system/framework. The key elements for both methods include team members’ inputs to plan and perform the work. On the one hand, no fixed role is defined in LPS, and the process relies on the “last planner”. On the other hand, the Scrum framework description clearly defines three main roles and assigns responsibilities for maintaining Scrum artifacts. For instance, the Scrum team collaborates in refining the Product Backlog, but the ultimate responsibility of its content and organization lies with the Product Owner. Likewise, the Development Team is responsible for managing the Sprint Backlog.

In LPS, the last planner acts as the system implementation facilitator. This facilitator changes over the course of the project. For example, during design, the last planner can be a senior architect. Later, during construction, the last planner can be a project manager or senior planner. In Scrum, the framework facilitator process lies in the Scrum Master. The Scrum Master also ensures that the team applies the Scrum framework, practices, and rules consistently throughout the entire project. In LPS, the “system or rule keeper” role may be adopted by one of the last planners. However, this role is not explicitly described in the system.

In terms of team size, LPS does not define or

recommend a specific number of team members. In Scrum, the team should be “small enough to remain nimble and large enough to complete significant work within a Sprint” (Schwaber and Sutherland, 2013). A small team may lack the skills and resources to function correctly. A large team (more than nine) requires great coordination to manage an empirical process.

(6) Regular events or team meetings: LPS and Scrum have relatively similar events. Planning events to derive the LPS master and phase schedule are equivalent to the Scrum Product Planning and Release Planning. In the Scrum Product Planning, the team envisions the main product features and defines the delivery sequence in the Release Plan. The WWP planning and review in the LPS are equivalent to the Sprint planning and review in Scrum. The root cause analysis process in the WWP is equivalent to Scrum Retrospective. Here, LPS and Scrum teams seek to capture learning for improving the planning and execution of upcoming work. LPS and Scrum also have daily events (e.g., Daily huddles) for rapid team feedback.

(7) Metrics/Dashboards: LPS and Scrum use several metrics to track project progress and performance. Both methods also encourage the use of visual controls and dashboards to consolidate relevant project information and ensure its transparency to project stakeholders.

The main difference lies in the way metrics are calculated. In LPS, a task is the key unit to derive most metrics. Accordingly, the ways the team sizes and screens activities through the different planning levels have significant effects on the final metrics’ values. Arguably, LPS team members have less influence on further weighting activities. By contrast, Scrum teams dedicate more time to collaboratively define and weigh tasks on the basis of an amalgam of experience, complexity, risk, and value delivered to customers.

(8) Approach to learning: In LPS, the team analyzes the Frequency of plan failures to address the root cause of issues and adopt corresponding countermeasures. LPS recommends the use of methods, such as 5 Whys, Plan–Do–Check–Act (PDCA), and Detect–Correct–Analyze–Prevent (DCAP), to systematically anticipate, react, and learn from breakdowns. Similarly, in Scrum, the team uses Sprint Retrospective to inspect how the last Sprint performed in terms of analysis of people, processes, and tools. On this basis, the team creates a plan to improve the way the team works. In LPS and Scrum, learning is also embraced in rapid team feedback derived from the work that is performed, and the team can make adaptations to the work on the basis of that feedback.

3 Discussion and recommendations

The authors discuss what Scrum elements can be potentially used to improve the LPS on the basis of the eight dimensions. Table 2 summarizes the recommendations.

Table 2 Summary of the main recommendation for Scrum elements to be implemented in LPS

Dimension	Main difference	Recommendation for LPS
Origins	Not applicable	—
Main purpose	Not applicable	—
Overall system/framework process	No significant difference	—
Tools or artifacts maintained by the team	Scrum Increment concept not addressed in the LPS	Explore the use of the concept of Scrum Increment during design
Team composition and main roles	Scrum offers a robust description of roles and responsibilities	Improve role description Add a Scrum Master
Regular events or team meetings	Scrum teams are arguably self-organized	Explore working with decentralized teams and Scaled Agile
Metrics/Dashboards	Scrum uses story points based on customer value	Explore the use of story points in LPS metrics
Approach to learning	No significant difference	—

1) Tools or artifacts maintained by the team: Explore the use of Scrum Increment concept in design

A Scrum Increment is a product functionality that is developed by the team during each Sprint. Increment is defined as “potentially shippable or of use to the Product Owner’s stakeholders” (Sutherland and Schwaber, 2007). This concept is not explicitly covered in the LPS.

At the construction phase, the LPS team works in the Weekly Work Plan (Sprint), but at the end of the week, they do not necessarily release a “usable” slice of the product (building). Buildings or infrastructure projects are complex products that can hardly be used before they are fully completed. Although the team can define a “usable increment”, this work will likely take a couple of weeks (or Sprints) to complete. Accordingly, the Increment concept may hardly be applicable during a project construction phase. However, the Increment may have potential applicability during the design phase of the project. Here, an Increment can be a finalized portion of the project design that can be “usable” to release other activities in the project that depends on the finalization of the design. The LPS team then should think about the design work as a series of small increments toward the end-design-product and formalize the conditions of satisfaction to consider an increment as “released”.

2) Team composition and main roles: Improve role description and add Scrum Master

A documented challenge in the implementation of LPS is the ambiguity in roles and responsibilities (Dave et al., 2015). The term “last planner” aims to emphasize collaboration in the planning process to have the inputs from people that know how to do the work efficiently. However, this concept seems to be misleading to teams implementing LPS. In many cases, one last planner ends up carrying the load of LPS implementation at the different planning levels, and this situation leads to partial applications of the system. By contrast, the Scrum framework defines roles and responsibilities, and who in the team is responsible for the different tools/artifacts used. LPS may benefit from using Scrum as a benchmark to

define roles and responsibilities.

The Scrum Master role as a “rule keeper” is also not explicitly covered in the LPS. As per the authors’ experience in LPS implementation, this role is always taken by a team member with previous experience in the implementation of the LPS. In other cases, some external consultants are brought to the team. In any case, having someone in the team to look into the implementation of the LPS may facilitate and guarantee the system’s implementation.

3) Regular events or team meetings: Explore working with decentralized teams and Scaled Agile

In LPS, there is a defined structure for organizing the work. Upon sound and well-defined tasks screened through the different planning levels, the last planners decide what “made ready” tasks should be carried out in each week (Koskela and Howell, 2002). Arguably, the structure to organize the work in Scrum teams is more self-organized than that in the LPS. The Development Team decides what tasks to do in interaction with the rest of the team and stakeholders on the basis of the changing customer requirements. This structure allows the Scrum team to have the flexibility to embrace change as they move in project completion.

The self-organization aspect in Scrum works well in co-located small teams, but it becomes counterproductive as the team grows. Practitioners have developed concepts, such as Scaling Scrum, wherein the original framework is adjusted to fit into large and decentralized teams, to overcome the challenges of implementing Scrum in a large setup (Schwaber, 2007). Scale Scrum implementation is not as mature as the original framework, and several challenges still prevail (Paasivaara and Lassenius, 2011; 2016). However, the aspects of Scale Scrum frameworks (e.g., Large-Scale Scrum (LeSS) (Larman and Vodde, 2010)) that can be leveraged to improve LPS are worth being explored, which can contribute to realizing LPS adaptations to fit offsite teams working in decentralized AEC projects (Ballard and Tommelein, 2016). A balance between centralization and decentralization on the ways

team operates should be achieved to enable the usability of the framework when implemented in AEC projects.

4) Metrics/Dashboards: Use of story points

The Scrum concept of story points is not explicitly covered in the LPS. LPS uses percentages to calculate metrics, but its relative significance is not adequately weighted. By contrast, Scrum teams have several guidelines to help teams estimate the effort of their task on the basis of a mixture of parameters, such as personal experience, complexity, risk, and the value for the customer (e.g., planning poker). Scrum bases its metrics in the relative weight of tasks. This approach allows the Development Team to track project progress on the basis of the value delivered to the customer instead of focusing on the number of internal resources. These metrics can be explored in more detail to analyze how they can be incorporated into LPS to improve the way the team measures themselves against project progress and also in terms of the value delivered to the customer.

4 Conclusions and directions for future research

The AEC industry is evolving toward integrated forms of project delivery, and engineering of modern infrastructure is becoming increasingly complex, which requires a re-evaluation and adaptation of traditional AEC practices. Looking into what other industries are doing and learning from their experiences can also help those in the AEC industry seeking improved ways to deliver modern infrastructure. In this article, LPS and Scrum are compared to find what elements of Scrum can potentially be used to improve the current LPS benchmark.

In general, LPS and Scrum share several principles related to the way the teams collaborate to organize the work and increase the value delivered to the customer. After analyzing LPS and Scrum's origins, purpose, system/framework processes, tools/artifacts, team composition, roles, events, meetings, metrics, and approach to learning side by side, the authors identified the following four main elements from Scrum that can be leveraged to improve the LPS benchmark.

1) Implement the concept of Scrum Increment into LPS, particularly when using LPS in the design phase. This may contribute to coping with the increased uncertainty, speed, and complexity inherent to the iterative design process.

2) Use Scrum as a benchmark for the LPS process description to clearly define roles and responsibilities. Having an equivalent to a Scrum Master as designated "rule keeper" in LPS could contribute to addressing some LPS challenges related to ambiguity in planning roles and responsibilities while ensuring consistency and continuity in the implementation of LPS across the different phases of construction projects.

3) Research beyond the basic Scrum framework and

exploration of concepts, such as Scale Scrum, to see how LPS can benefit from these expanded applications. This may help in finding ways to incorporate offsite or remote teams into the LPS.

4) Explore the use of the Scrum story points into existing LPS metrics. This can complement current LPS metrics in terms of consistency and correlation to the overall team and project performance.

It is important to note that Scrum has been used mainly in the information systems industry, which differs from the construction industry in terms of project scope, workforce, resources, and risk, to name a few. Therefore, embedding Scrum practices into the LPS requires transferring the concepts considering project contexts. The recommendations proposed here should be further explored and tested in real projects to evaluate their value. Researchers in the information systems industry may also be interested in investigating the transfer of LPS practices into Scrum or in IT projects.

References

- Abbas N, Gravell A M, Wills G B (2008). Historical roots of Agile methods: Where did "Agile thinking" come from? In: International Conference on Agile Processes and Extreme Programming in Software Engineering. Berlin, Heidelberg: Springer, 94–103
- AIA (2007). Integrated Project Delivery: A Guide. Version 1. The American Institute of Architects (AIA)
- Ballard G (2000). The Last Planner System of Production Control. Dissertation for the Doctoral Degree. Birmingham: University of Birmingham
- Ballard G, Hammond J, Nickerson R (2009). Production control principles. In: Proceedings of the 17th Annual Conference of the International Group for Lean Construction (IGLC). Taipei, 489–500
- Ballard G, Howell G (1998). Shielding production: Essential step in production control. *Journal of Construction Engineering and Management*, 124(1): 11–17
- Ballard G, Howell G, Casten M (1996). PARC: A case study. In: Proceedings of the 4th Annual Conference of the International Group for Lean Construction (IGLC). Birmingham
- Ballard G, Howell G A (2003a). Competing construction management paradigms. In: Construction Research Congress: Wind of Change—Integration and Innovation, 1–8
- Ballard G, Howell G A (2003b). An update on last planner. In: Proceedings of the 11th Annual Conference of the International Group for Lean Construction (IGLC). Blacksburg, VA
- Ballard G, Tommelein I D (2016). Current process benchmark for the last planner system. *Lean Construction Journal*, 89: 57–89
- Barlow J (1998). From craft production to mass customisation? Customer-focused approaches to housebuilding. In: Proceedings of the 6th Annual Conference of the International Group for Lean Construction (IGLC). Sao Paulo
- Beck K, Beedle M, van Bennekum A, Cockburn A, Cunningham W, Fowler M, Grenning J, Highsmith J, Hunt A, Jeffries R, Kern J, Marick B, Martin R C, Mellor S, Schwaber K, Sutherland J, Thomas

- D (2001). *Manifesto for Agile Software Development*
- Boyes H A (2013). Cyber security of intelligent buildings: A review. In: 8th IET International System Safety Conference incorporating the Cyber Security Conference. Cardiff
- Caballero E, Calvo-Manzano J A, San Feliu T (2011). Introducing scrum in a very small enterprise: A productivity and quality analysis. In: O'Connor R V, Pries-Heje J, Messnarz R, eds. *European Conference on Software Process Improvement. Systems, Software and Service Process Improvement*. Roskilde: Springer, 215–224
- Cardozo E S F, Araújo Neto J B F, Barza A, França A C C, da Silva F Q B (2010). SCRUM and productivity in software projects: A systematic literature review. In: 14th International Conference on Evaluation and Assessment in Software Engineering (EASE). Keele University, 1–4
- Cervone H F (2011). Understanding Agile project management methods using Scrum. *OCLC Systems & Services: International Digital Library Perspectives*, 27(1): 18–22
- Ciric D, Lalic B, Gracanin D, Palcic I, Zivlak N (2018). Agile project management in new product development and innovation processes: Challenges and benefits beyond software domain. In: *IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)*. Beijing, 1–9
- Cohn M (2005). *Agile Estimating and Planning*. Englewood: Prentice Hall
- Cohn M (2014). My primary criticism of scrum. Available at: mountaingoatsoftware.com/blog
- Dave B, Boddy S, Koskela L (2011). VisiLean: Designing a production management system with lean and BIM. In: *Proceedings of the 19th Annual Conference of the International Group for Lean Construction (IGLC)*. Lima, 477–487
- Dave B, Boddy S, Koskela L (2013). Challenges and opportunities in implementing lean and BIM on an infrastructure project. In: *Proceedings of the 21st Annual Conference of the International Group for Lean Construction (IGLC)*. Fortaleza, 741–750
- Dave B, Hämäläinen J P, Koskela L (2015). Exploring the recurrent problems in the last planner implementation on construction projects. In: *Proceedings of the Indian Lean Construction Conference (ILCC 2015)*. Institute for Lean Construction Excellence
- Demir S T, Theis P (2016). Agile design management—The application of Scrum in the design phase of construction projects. In: *Proceedings of the 24th Annual Conference of the International Group for Lean Construction*. Boston, MA, 13–22
- El Samad G, Hamzeh F R, Emdanat S (2017). Last planner system—The need for new metrics. In: *Proceedings of the 25th Annual Conference of the International Group for Lean Construction (IGLC)*. Heraklion, 637–644
- Fischer M, Ashcraft H W, Reed D, Khanzode A (2017). *Integrating Project Delivery*. Hoboken: John Wiley & Sons
- Fisk D (2012). Cyber security, building automation, and the intelligent building. *Intelligent Buildings International*, 4(3): 169–181
- García de Soto B, Adey B T, Fernando D (2017). A hybrid methodology to estimate construction material quantities at an early project phase. *International Journal of Construction Management*, 17(3): 165–196
- García de Soto B, Agustí-Juan I, Joss S, Hunhevicz J (2019). Implications of construction 4.0 to the workforce and organizational structures. *International Journal of Construction Management*, 1–13
- Gonzalez V, Alarcon L F, Mundaca F (2008). Investigating the relationship between planning reliability and project performance. *Production Planning and Control*, 19(5): 461–474
- Grenning J (2002). Planning poker or how to avoid analysis paralysis while release planning. Hawthorn Woods: Renaissance Software Consulting, 3: 22–23
- Hajdu M (2018). Survey of precedence relationships: Classification and algorithms. *Automation in Construction*, 95: 245–259
- Hajdu M, Lucko G, Su Y (2017). Singularity functions for continuous precedence relations and nonlinear activity-time-production functions. *Automation in Construction*, 79: 31–38
- Hamdi O, Leite F (2012). BIM and Lean interactions from the BIM capability maturity model perspective: A case study. In: *Proceedings of the 20th Annual Conference of the International Group for Lean Construction (IGLC)*. San Diego, CA
- Hamzeh F R, Aridi O Z (2013). Modeling the last planner system metrics: A case study of an AEC company. In: *Proceedings of the 21st Annual Conference of the International Group for Lean Construction (IGLC)*. Fortaleza, 599–608
- Hamzeh F R, Ballard G, Tommelein I D (2009). Is the Last Planner System applicable to design? A case study. In: *Proceedings of the 17th Annual Conference of the International Group for Lean Construction (IGLC)*. Taipei, 165–176
- Heigermoser D, García de Soto B, Abbott E L S, Chua D K H (2019). BIM-based Last Planner System tool for improving construction project management. *Automation in Construction*, 104: 246–254
- Highsmith J, Cockburn A (2001). Agile software development: The business of innovation. *Computer*, 34(9): 120–127
- Hossain E, Babar M A, Paik H Y (2009). Using scrum in global software development: A systematic literature review. In: *4th IEEE International Conference on Global Software Engineering*. Limerick, 175–184
- Iamandi O, Popescu S, Dragomir M, Morariu C (2015). A critical analysis of project management models and its potential risks in software development. *Calitatea*, 16(149): 55–61
- Kalsaas B T, Bonnier K E, Ose A O (2016). Towards a model for planning and controlling ETO design projects. In: *Proceedings of the 24th Annual Conference of the International Group for Lean Construction (IGLC)*. Boston, MA, 33–42
- Khan S, Tzortzopoulos P (2015). Improving design workflow with the last planner system: Two action research studies. In: *Proceedings of the 23rd Annual Conference of the International Group for Lean Construction (IGLC)*. Perth, 568–577
- Koskela L (1992). *Application of the new production philosophy to construction*. CIFE Technical Report #72. Stanford: Stanford University
- Koskela L (2000). *An Exploration towards a Production Theory and Its Application to Construction*. Dissertation for the Doctoral Degree. Espoo: VTT Technical Research Centre of Finland
- Koskela L, Ballard G, Tanhuanpää V P (1997). Towards lean design management. In: *Proceedings of the 5th Annual Conference of the International Group for Lean Construction (IGLC)*. Gold Coast, 1–13
- Koskela L, Howell G (2002). The theory of project management: Explanation to novel methods. In: *Proceedings of the 10th Annual Conference of the International Group for Lean Construction (IGLC)*.

- Gramado, 1–11
- Kracfik J F (1988). Triumph of the Lean production system. *MIT Sloan Management Review*, 30(1): 41
- Kupp M, Dahlander L, Morrow E (2015). Team Wikispeed: Developing hardware the software way. Case Study. *European School of Management and Technology*
- Larman C, Vodde B (2010). *Practices for Scaling Lean & Agile Development: Large Multisite and Offshore Product Development with Large-Scale Scrum*. Upper Saddle River, NJ: Addison-Wesley Professional
- Lessing J, Stehn L, Ekholm A (2005). Industrialised housing: Definition and categorization of the concept. In: *Proceedings of the 13th Annual Conference of the International Group for Lean Construction (IGLC)*. Sydney, 471–480
- Lia K A, Ringerike H, Kalsaas B T (2014). Increase predictability in complex engineering and fabrication projects. In: *Proceedings of the 22nd Annual Conference of the International Group for Lean Construction (IGLC)*. Oslo, 437–449
- Liu M, Ballard G (2008). Improving labor productivity through production control. In: *Proceedings of the 16th Annual Conference of the International Group for Lean Construction (IGLC)*. Manchester, 657–666
- Liu M, Ballard G, Ibbs W (2011). Work flow variation and labor productivity: Case study. *Journal of Management Engineering*, 27(4): 236–242
- Lu W, Olofsson T, Jensen P, Simonsson P (2011). BIM-based lean-agile supply chain for industrialized housing. In: *International Conference on Construction Applications of Virtual Reality*. Bauhaus-Universität Weimar, 262–270
- Mantha B, García de Soto B (2019). Cyber security challenges and vulnerability assessment in the construction industry. In: *Proceedings of the 2019 Creative Construction Conference*. Budapest, 29–37
- Mostafa S, Chileshe N, Abdelhamid T (2016). Lean and agile integration within offsite construction using discrete event simulation: A systematic literature review. *Construction Innovation*, 16(4): 483–525
- Naim M, Naylor J, Barlow J (1999). Developing lean and agile supply chains in the UK housebuilding industry. In: *Proceedings of the 7th Annual Conference of the International Group for Lean Construction (IGLC)*. University of California, Berkeley, CA, 159–170
- Ormeño Zender Y, García de Soto B (2020). Use of Scrum in the rehabilitation of a commercial building in Peru. *Construction Innovation: Information, Process, Management*. In press, doi: 10.1108/CI-12-2019-0140
- Oskouie P, Gerber D J, Alves T, Becerik-Gerber B (2012). Extending the interaction of building information modeling and lean construction. In: *Proceedings of the 20th Annual Conference of the International Group for Lean Construction (IGLC)*. San Diego, CA, 111–120
- Owen R L, Koskela L (2006). Agile construction project management. In: *6th International Postgraduate Research Conference in the Built and Human Environment*, 6(7): 22–33
- Owen R L, Koskela L J, Henrich G, Codinhoto R (2006). Is Agile project management applicable to construction? In: *Proceedings of the 14th Annual Conference of the International Group for Lean Construction (IGLC)*. Pontificia Universidad Católica de Chile, Santiago, 51–66
- Paasivaara M, Lassenius C (2011). Scaling Scrum in a large distributed project. In: *International Symposium on Empirical Software Engineering and Measurement*. Banff, 363–367
- Paasivaara M, Lassenius C (2016). Scaling scrum in a large globally distributed organization: A case study. In: *IEEE 11th International Conference on Global Software Engineering (ICGSE)*. Irvine, CA, 74–83
- Parn E A, Edwards D (2019). Cyber threats confronting the digital built environment: Common data environment vulnerabilities and block chain deterrence. *Engineering, Construction, and Architectural Management*, 26(2): 245–266
- Royce W W (1970). Managing the development of large software systems. In: *Proceedings of the 9th International Conference on Software Engineering*. IEEE WESCON, 328–338
- Rubin K S (2012). *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Upper Saddle River, NJ: Addison-Wesley
- Sacks R, Barak R, Belaciano B, Gurevich U, Pikas E (2013). KanBIM workflow management system: Prototype implementation and field testing. *Lean Construction Journal*, 19–35
- Sacks R, Koskela L, Dave B, Owen R (2010). Interaction of lean and building information modeling in construction. *Journal of Construction Engineering and Management*, 136(9): 968–980
- Sanchez L M, Nagi R (2001). A review of Agile manufacturing systems. *International Journal of Production Research*, 39(16): 3561–3600
- Sanvido V E, Konchar M (1999). Selecting project delivery systems: Comparing design-build, design-bid-build and construction management at risk. *Project Delivery Institute*, State College, PA
- Schwaber K (2007). *The Enterprise and Scrum*. Redmond, WA: Microsoft Press
- Schwaber K, Beedle M (2001). *Agile Software Development with Scrum*. Upper Saddle River, NJ: Prentice Hall
- Schwaber K, Sutherland J (2013). *The Scrum Guide™: The definitive guide to Scrum—The rules of the game*
- Sriprasert E, Dawood N (2003). Multi-constraint information management and visualization for collaborative planning and control in construction. *Journal of Information Technology in Construction*, 8: 341–366
- Streule T, Miserini N, Bartlomé O, Klippel M, García de Soto B (2016). Implementation of scrum in the construction industry. *Procedia Engineering*, 164: 269–276
- Sutherland J, Schwaber K (2007). *The Scrum Papers: Nuts, Bolts and Origins of an Agile Process*. Cambridge: Scrum Inc.
- Takeuchi H, Nonaka I (1986). The new product development game. *Harvard Business Review*, 64(1): 137–146
- Tiwari S, Sarathy P (2012). Pull planning as a mechanism to deliver constructible design. In: *Proceedings of the 20th Annual Conference of the International Group for Lean Construction (IGLC)*. San Diego, CA, 18–20
- Tommelein I D, Ballard G (1997). Look-ahead planning: Screening and pulling. Technical Report No. 97-9. *Construction Engineering and Management Program, Civil and Environmental Engineering Department, University of California, Berkeley, CA*
- Womack J P, Jones D T, Roos D (1990). *The Machine that Changed the World: The Story of Lean Production*. New York: Rawson Associates