Edge Computing Benefits in Low-Latency IoT Applications: A Comprehensive Survey

Prof. Claudio Enrico Palazzi   
*Department of Mathematics*

*University of Padua*

*Padua, Italy*  
cpalazzi@math.unipd.it

Michael Amista’  
*Department of Mathematics*

*University of Padua*

*Padua, Italy*  
michael.amista@studenti.unipd.it

*Abstract*—The rapid proliferation of Internet of Things (IoT) devices has generated unprecedented volumes of data, creating significant challenges in computational processing, latency, and resource management. This comprehensive survey explores the transformative role of edge computing in addressing critical limitations of traditional cloud-based IoT architectures. By examining recent advances in edge computing technologies, this work provides insights on how computational capabilities can be extended from centralized cloud infrastructure to the network's edge, enabling more efficient, responsive, and intelligent IoT applications.

The investigation reveals that edge computing fundamentally reshapes IoT capabilities by bringing computational resources closer to data sources, thereby mitigating network bandwidth constraints and communication latency. The survey systematically analyses edge computing architectures across diverse domains, including mobile applications, industrial manufacturing, and interactive technologies. Particular emphasis is placed on mobile edge computing (MEC) paradigm that addresses energy efficiency, computation offloading, and low-latency requirements. Key findings demonstrate edge computing's potential to support real-time processing, enhance computational autonomy, and optimize resource utilization for resource-constrained smart devices.

Furthermore, the survey identifies persistent research challenges and opportunities, highlighting the need for continued innovation in edge computing infrastructure, communication protocols, and intelligent resource management strategies. By providing a comprehensive overview of current technological landscapes, this study offers researchers and practitioners a critical reference for understanding and implementing edge computing solutions in IoT ecosystems.

Keywords—Edge computing, Internet of Things (IoT), Low-Latency Applications, Mobile game, IoT-Based Manufacturing, Mobile Edge Computing (MEC), Real-Time Processing, Energy Efficiency

# Introduction: context and study cases

The Internet of Things (IoT) represents a transformative technological paradigm that has rapidly evolved over the past decade. At its core, IoT encompasses a vast network of interconnected devices equipped with sensors, software, and network connectivity, enabling them to collect, exchange, and process data. From smart home devices and wearable technologies to industrial sensors and urban infrastructure monitoring systems, IoT has emerged as a critical technological ecosystem that bridges the physical and digital worlds.

The proliferation of IoT devices has been exponential, with global estimates suggesting over 75 billion connected devices by 2025. These devices range from simple sensors measuring environmental parameters to complex systems managing critical infrastructure. However, the initial generation of IoT devices was predominantly limited to data collection and transmission, with minimal on-site computational capabilities.

## Limitations of traditional cloud computing

Traditional cloud computing models have been the primary approach for processing IoT-generated data. In this centralized paradigm, devices collect data and transmit it to remote data centres for processing and analysis. While this approach worked effectively for early IoT applications, it has become increasingly inadequate for emerging low-latency use cases.

The primary limitations of traditional cloud computing in IoT contexts include:

* **Network Bandwidth Constraints**: Transmitting large volumes of data from numerous devices to centralized cloud servers creates significant network congestion and bandwidth challenges.
* **Communication Latency**: The physical distance between IoT devices and cloud data centres introduces substantial processing delays, rendering the approach unsuitable for low-latency applications.
* **Resource Inefficiency**: Sending all collected data to remote servers for processing is computationally and energetically inefficient, especially for resource-constrained devices.
* **Privacy and Security Concerns**: Continuous data transmission to external servers raises critical questions about data privacy and potential security vulnerabilities.

## Emergence of edge computing

Edge computing has emerged as a revolutionary solution to address the inherent limitations of traditional cloud-based IoT architectures. By bringing computational capabilities closer to the data sources, directly at the network's edge, this paradigm fundamentally transforms how IoT systems process and analyse information.

Key characteristics of edge computing include:

* Distributed computational resources
* Local data processing and filtering
* Reduced latency
* Enhanced real-time decision-making capabilities
* Improved energy efficiency
* Increased system autonomy

The technological advancements in embedded systems have been fundamental in enabling edge computing. Modern IoT devices now possess sufficient computational resources to perform complex data processing tasks locally, marking a significant departure from earlier generations of limited-capability sensors.

## Paper scope and contribution

This comprehensive survey provides a systematic exploration of edge computing technologies within the IoT ecosystem, addressing critical computational challenges in modern low-latency IoT applications.

The survey is structured around three primary research dimensions. First, it examines the architectural transformation from centralized cloud computing to distributed edge computing architectures. This exploration investigates technological advancements that enable more efficient, responsive, and intelligent IoT systems by bringing computational capabilities closer to data sources.

Second, the research systematically explores edge computing implementations across multiple technological domains. These include mobile applications, industrial manufacturing, interactive technologies, and mobile edge computing frameworks. By analysing these diverse contexts, the survey provides a holistic understanding of edge computing's versatility and potential.

Third, the survey emphasizes performance optimization by highlighting how edge computing addresses fundamental limitations in IoT systems. This includes strategies for mitigating network bandwidth constraints, reducing communication latency, supporting real-time processing, enhancing computational autonomy, and optimizing resource utilization for resource-constrained devices.

At the end, this works provides insights on current research challenges and opportunities to promote the continuous investigation and innovation in edge computing infrastructures.

# edge computing in internet of things

## Role of edge computing

Edge computing emerges as a strategic technological paradigm that fundamentally transforms how Internet of Things (IoT) systems process and interact with data. As highlighted by Hassan et al. [1], edge computing plays multiple critical roles in IoT ecosystems:

**Data Acquisition and Processing**: Edge devices, including sensors and intelligent machines, now possess the capability to capture streaming data and perform immediate analysis. This approach aligns with the evolving computational philosophy of "moving the algorithm to the data" rather than transporting data to centralized algorithms. For instance, in smart transportation systems, traffic light cameras can simultaneously capture and analyse data, enabling instantaneous decision-making to optimize traffic flow.

**Inferential Controls**: Edge devices are increasingly equipped with sophisticated inferential capabilities, allowing them to interpret environmental contexts accurately. These devices can communicate with broader infrastructural systems while making intelligent, contextually-aware decisions. In smart transportation scenarios, this translates to providing drivers with highly intelligent navigation instructions by integrating data from GPS and multiple camera inputs.

**Real-Time Data Analysis**: By enabling localized data analysis at the point of generation, edge computing significantly reduces information latency. This approach offers multiple advantages:

• Faster generation of actionable insights

• Reduced network bandwidth consumption

• Decreased operational costs

• Immediate decision-making capabilities

Across industries such as manufacturing, healthcare, telecommunications, and finance, edge computing facilitates more efficient and responsive IoT implementations.

**Localized Decision Making**: Edge devices can now process and make critical strategic decisions locally, eliminating the need for time-consuming data transmission to centralized cloud infrastructure. In scenarios like smart transportation, where milliseconds matter, such as autonomous vehicle systems, local data processing becomes crucial for real-time safety and performance.

**Enhanced Data Security**: By localizing data collection and analysis, edge computing inherently improves data security. Reduced extensive routing minimizes potential vulnerability points, making it easier to identify and mitigate suspicious activities before they escalate into significant security breaches. This multifaceted approach positions edge computing as a transformative technology that addresses fundamental limitations in traditional IoT architectures, offering unprecedented levels of efficiency, responsiveness, and intelligence.

## Key requirements for successful deployment

As discussed by Hassan et al. [1], successful deployment of edge computing in IoT environment should meet specific requirements to achieve all edge computing features. Since several of these requirements are conflicting, application designers must find a good balance among all of them. These key requirements are:

**Latency**: Edge computing addresses the delay of traditional cloud models by processing data closer to its source, enabling real-time responsiveness. This is critical for applications like healthcare, autonomous vehicles, and industrial automation, where even minimal delays can have severe consequences.

**Reliability**: Reliable edge systems ensure consistent performance with minimal downtime, even in diverse and challenging conditions. This includes maintaining computational integrity and service quality while integrating with varied IoT ecosystems.

**Mobility Support**: As IoT devices increasingly operate on the move, edge computing must provide seamless connectivity and uninterrupted service. Robust mobility support ensures smooth handoffs, session continuity, and reliable performance in dynamic network environments.

**Real-time interactions**: Many IoT applications require immediate data processing and response. Edge computing supports real-time interactions essential for systems like collision avoidance in autonomous vehicles and industrial monitoring.

**Security**: Distributed edge systems introduce multiple vulnerability points. Comprehensive security measures, including data protection and continuous threat monitoring, are vital to safeguard against attacks while leveraging localized data processing.

**Interoperability**: Edge computing must integrate seamlessly across diverse hardware, protocols, and applications. Standardized interfaces and adaptive frameworks enable effective communication and collaboration in complex IoT ecosystems.

## Applications

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

# Prepare Your Paper Before Styling

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

## Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

## Units

* Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as “3.5-inch disk drive”.
* Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
* Do not mix complete spellings and abbreviations of units: “Wb/m2” or “webers per square meter”, not “webers/m2”. Spell out units when they appear in text: “. . . a few henries”, not “. . . a few H”.
* Use a zero before decimal points: “0.25”, not “.25”. Use “cm3”, not “cc”. (*bullet list*)

## Equations

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

Number equations consecutively. Equation numbers, within parentheses, are to position flush right, as in (1), using a right tab stop. To make your equations more compact, you may use the solidus ( / ), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in:

*a**b* 

Note that the equation is centered using a center tab stop. Be sure that the symbols in your equation have been defined before or immediately following the equation. Use “(1)”, not “Eq. (1)” or “equation (1)”, except at the beginning of a sentence: “Equation (1) is . . .”

## Some Common Mistakes

* The word “data” is plural, not singular.
* The subscript for the permeability of vacuum **0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter “o”.
* In American English, commas, semicolons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
* A graph within a graph is an “inset”, not an “insert”. The word alternatively is preferred to the word “alternately” (unless you really mean something that alternates).
* Do not use the word “essentially” to mean “approximately” or “effectively”.
* In your paper title, if the words “that uses” can accurately replace the word “using”, capitalize the “u”; if not, keep using lower-cased.
* Be aware of the different meanings of the homophones “affect” and “effect”, “complement” and “compliment”, “discreet” and “discrete”, “principal” and “principle”.
* Do not confuse “imply” and “infer”.
* The prefix “non” is not a word; it should be joined to the word it modifies, usually without a hyphen.
* There is no period after the “et” in the Latin abbreviation “et al.”.
* The abbreviation “i.e.” means “that is”, and the abbreviation “e.g.” means “for example”.

An excellent style manual for science writers is [7].

# Using the Template

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

## Authors and Affiliations

**The template is designed for, but not limited to, six authors.** A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

### For papers with more than six authors: Add author names horizontally, moving to a third row if needed for more than 8 authors.

### For papers with less than six authors: To change the default, adjust the template as follows.

#### Selection: Highlight all author and affiliation lines.

#### Change number of columns: Select the Columns icon from the MS Word Standard toolbar and then select the correct number of columns from the selection palette.

#### Deletion: Delete the author and affiliation lines for the extra authors.

## Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

## Figures and Tables

#### Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead | Subhead | Subhead |
| copy | More table copya |  |  |

1. Sample of a Table footnote. (*Table footnote*)
2. Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

1. N. Hassan, S. Gillani, E. Ahmed, I. Yaqoob and M. Imran, "The Role of Edge Computing in Internet of Things," in IEEE Communications Magazine, vol. 56, no. 11, pp. 110-115, November 2018, doi: 10.1109/MCOM.2018.1700906.
2. G. Premsankar, M. Di Francesco and T. Taleb, "Edge Computing for the Internet of Things: A Case Study," in IEEE Internet of Things Journal, vol. 5, no. 2, pp. 1275-1284, April 2018, doi: 10.1109/JIOT.2018.2805263.
3. B. Chen, J. Wan, A. Celesti, D. Li, H. Abbas and Q. Zhang, "Edge Computing in IoT-Based Manufacturing," in IEEE Communications Magazine, vol. 56, no. 9, pp. 103-109, Sept. 2018, doi: 10.1109/MCOM.2018.1701231.
4. K. Zhang, S. Leng, Y. He, S. Maharjan and Y. Zhang, "Mobile Edge Computing and Networking for Green and Low-Latency Internet of Things," in IEEE Communications Magazine, vol. 56, no. 5, pp. 39-45, May 2018, doi: 10.1109/MCOM.2018.1700882.