Metric Temporal Logic (MTL) extends Linear Temporal Logic (LTL) by introducing temporal operators with explicit time constraints, such as “Finally within interval I” (F\_I) and “Globally within interval I” (G\_I). This allows for precise descriptions of system behaviors over time, making it particularly useful for the specification and verification of real-time and hybrid systems. For example, MTL can specify that a response must occur within 2 seconds after detecting an obstacle (F\_{(0,2)} response) or that a system must always maintain a safe distance while operating (G\_{(0,\infty)} safe\_distance). (Havelund and Peled, 2019; Ashari and Habib, 2023).

References:

* Havelund, K. and Peled, D. (2019) ‘An Extension of LTL with Rules and its Application to Runtime Verification’, *Runtime Verification*, pp. 1-22.

度量时态逻辑（Metric Temporal Logic，简称MTL或MLTL）是一种扩展的时态逻辑，用于描述系统在特定时间范围内的行为。它在实时系统和混合系统的规范和验证中具有重要应用。

**基本概念**

MTL通过引入时间约束来扩展传统的时态逻辑。其基本运算符包括：

* **F\_I（Finally within interval I）**：表示在时间间隔I内某个条件最终会成立。
* **G\_I（Globally within interval I）**：表示在时间间隔I内某个条件始终成立。
* **U\_I（Until within interval I）**：表示某个条件会一直成立直到另一个条件在时间间隔I内成立。
* 在度量线性时态逻辑（MLTL）中的运算符（如F\_I）中，I表示一个时间间隔。这个时间间隔I定义了条件必须在多长时间内满足。例如，在F\_{(0,5)} p中，I是时间间隔(0,5)，表示条件p必须在未来0到5秒内最终成立。时间间隔I可以是闭区间、开区间或半开区间，具体取决于所需的时间约束。通过这种方式，MLTL能够精确描述系统在特定时间范围内的行为。

**语法和语义**

MTL公式的语法基于这些时间约束运算符。例如：

* **F\_{(0,5)} p**：表示条件p在未来0到5秒内最终会成立。
* **G\_{(0,10)} q**：表示条件q在未来0到10秒内始终成立。
* **p U\_{(0,3)} r**：表示条件p会一直成立直到条件r在未来0到3秒内成立。

[这些公式的语义通过在连续时间模型上进行评估来定义。MTL允许对系统行为进行精确的时间约束描述，这在实时系统的验证中尤为重要1](https://zhuanlan.zhihu.com/p/415602732)[2](https://link.springer.com/article/10.1007/s00236-015-0243-0)。

**应用**

MTL广泛应用于实时系统和混合系统的规范和验证。例如，在一个自动驾驶系统中，我们可以用MTL来描述以下性质：

* **响应时间**：在检测到障碍物后，系统必须在2秒内做出反应（F\_{(0,2)} response）。
* **安全性**：在行驶过程中，系统必须始终保持安全距离（G\_{(0,\infty)} safe\_distance）。

通过这些描述，可以确保系统在规定的时间范围内满足特定的行为要求，从而提高系统的可靠性和安全性。

**示例**

假设我们有一个温度控制系统，我们可以用MTL来描述以下性质：

* **温度调节**：温度在检测到过高后，必须在5秒内降到安全范围内（F\_{(0,5)} safe\_temperature）。
* **持续监控**：系统必须在整个运行过程中持续监控温度（G\_{(0,\infty)} monitor\_temperature）。

这些性质可以通过MTL公式来表达，并用于验证系统在运行过程中的行为是否符合预期。