arXiv:2603.15675v1 [cs.LO] 13 Mar 2026

构造弱终止接口协议

Debjyoti Bera[1]和 Tim A.C. Willemse[2]

[1] TNO-ESI，埃因霍温，荷兰
debjyoti.bera@tno.nl，[2] 埃因霍温理工大学，荷兰 t.a.c.willemse@tue.nl

摘要。 接口通过规定服务提供者（服务器）与其消费者（客户端）之间允许的交互，在确定兼容的组件组合方面发挥着核心作用。异步通信系统中的高度并发增加了意外引入死锁和活锁的风险。弱终止性质可作为一项基本的合理性检查，以避免此类问题。它保证在每个可达状态下，系统都有机会最终终止。本文推广了现有结果，这些结果通过构造保证接口组合弱终止。我们的推广使该理论在实践中具有更广泛的适用性。我们从满足特定性质的服务器接口规范出发，展示如何利用部分镜像关系推导出一类用于建模不同使用情境的客户端。此外，我们还讨论如何将这些结果嵌入开源工具，以指导建模者设计弱终止接口。

关键词： 接口 · 异步系统 · 弱终止

1 引言

基于组件的系统设计，第一步是将给定的系统设计问题分解为功能上协调一致的执行单元，即组件 [14,19]。从抽象模型出发，通过逐步细化进一步详细描述每个组件。为了实现共同目标，组件之间可能需要相互交互。在现代分布式系统中，这些交互通常通过消息传递以异步方式进行。组件对之间允许的交互集合定义了一种接口协议（契约），它包含两端：提供功能的服务器端和使用该功能的客户端。使用状态机形式化方法 [8] 来描述此类交互十分普遍。组件之间的接口定义完成后，通过组合满足某种预定义兼容性条件的共享客户端-服务器接口，即可构成最终系统。

基于组件的方法直接导致复杂性大幅增加，并发和异步通信使得

难以避免细微却关键的设计错误，从而导致死锁或活锁。事实证明，即使服务器端协议不存在这些问题，仅仅镜像该协议以获得定义客户端使用方式的客户端协议，也无法保证不会出现错误。因此，形式化建模与分析技术的使用势在必行。佩特里网为异步通信系统建模提供了一种自然的方式，并且有丰富的验证方法和工具支持 [2]。Petri网结构的图形化表示为系统建模提供了一种直观的方式。特别是，在验证系统行为时，结构分析技术克服了状态空间探索技术的缺点。

任何基于组件的系统都有一项基本要求：系统中的所有组件始终都应有机会到达所需的最终状态，并且所有已发送的消息都已被接收。此性质称为弱终止。弱终止并不意味着系统总会终止，而是意味着系统始终有机会终止。因此，该性质保证系统不会发生死锁或活锁。

对于某些 Petri 网子类，其结构蕴含弱终止，已有许多有趣的研究成果。在 [1] 中，提出了一种充分条件，用于成对验证树结构组合的弱终止性质。对于一种称为 ATIS网的组件对组合子类，该条件可由其结构直接推出 [11]。ATIS网由成对的无环标记图和同构状态机构成，并通过对成对库所进行同时细化而形成，所使用的细化对象是具有弱终止性质的多工作流网。在 [7],中，提出了一种基于组件和端口概念（用于定义通信协议）的架构框架。端口兼容性的充分条件以及组件组合的无环网络结构，能够使最终系统（由异步通信组件构成的网络）具有弱终止性质。在 [6],中，提出了另一种基于组件和交互模式概念的架构框架，其中交互模式是定义通信协议的参数化多工作流网。这些交互模式涵盖了实践中常见的三种模式，即远程过程调用、取消（目标-反馈-结果）和发布-订阅。在对现有系统进行的每次通信细化中，都会将一个新组件作为交互模式的一部分添加到系统中。在 [5], 中，架构框架 [6] 通过将兼容端口的概念从 [7] 推广到交互模式，扩展为包含镜像端口模式。

镜像端口模式定义了一类弱终止的单服务器与多个客户端交互模式，其中每个客户端都是服务器状态机的同构镜像，并满足一些结构性质，以确保符合诸如选择性质和腿性质等通信条件。前者保证发生冲突时，只有一方可以做出选择；后者则保证，分叉与汇合之间的每条路径都始终需要服务器和一个客户端参与。图 1 展示了镜像端口模式的一个示例。然而，事实证明，在实际应用中，镜像端口模式至少有以下三个方面的局限：

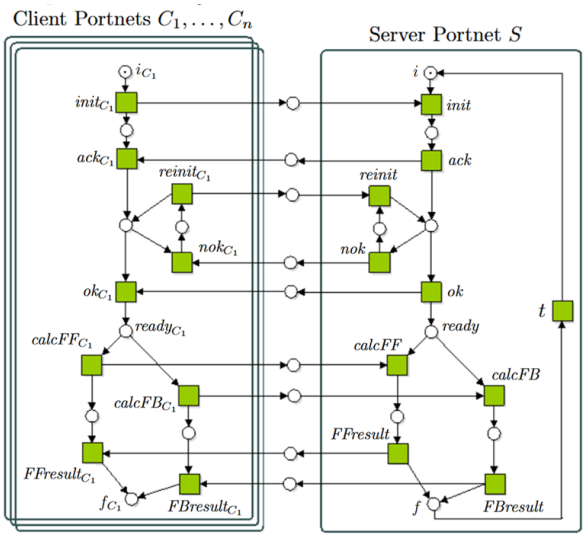


图 1：镜像端口网交互模式

(i) 选择性质要求服务器与客户端之间不存在竞争，也就是说，解决冲突变迁的主动权要么在服务器手中，要么在某个客户端手中，但不能双方都有。在实际应用中，这一限制相当严格，因为竞争往往是有意设计的。例如，在图 1 中，无法对以下竞争进行建模：客户端发射一条从就绪库所出发的变迁，同时服务器也可能从就绪库所发出一个错误。

(ii) 镜像端口网不允许服务器/客户端中的多个变迁连接到同一个接口库所。这意味着无法从多个状态发送/接收同一条消息。例如，在图 1 中，必须引入变迁重新初始化，因为变迁初始化 无法发生两次并共享同一个接口库所。

(iii) 同构要求意味着客户端端口网必须完整镜像服务器端口网，即使它们可能不会使用全部功能。例如，在图 1 中，所有客户端 C_1、.. . 、C_n 都必须实现从库所就绪出发的两种选择，即使某些客户端可能只会使用其中一种。这是一项常见要求，因为根据组件所处的上下文，每个客户端对服务器的使用方式可能各不相同。

本文重新审视了镜像端口交互模式 [5]，并通过添加标记函数对其进行扩展，放宽了对其结构的限制，并弱化了对端口网的同构要求。我们将这类交互模式称为部分镜像端口网，并证明这些模式具有弱终止性质。研究结果为设计和推理实践中遇到的大量接口的正确性提供了一种结构化方法。由于部分镜像端口网是基于组件的

框架中的一种交互模式，我们可以重复使用基于逐步细化的构造方法，从体系结构图出发，设计由异步通信组件构成的弱终止网络。这些研究结果已首次应用于开源的 ComMA 框架 [13]，该框架被业界工程师广泛用于规范软件接口。我们已实现相关算法，可自动检测接口规范中违反端口网性质的情况，并在设计时提供有关终止问题的反馈。

本文结构如下：第 2 节介绍基本的数学概念和符号。第 3 节介绍带标记的端口网、组合和部分镜像关系。第 4 节介绍带标记的端口网满足良构条件所需的限制。第 5 节证明，良构标记端口网与其部分镜像的组合具有弱终止性。第 6 节将这一结果推广至多个客户端。第 7 节展示这些结果在 ComMA 中的一项应用。本文在第 8 节作总结。

2 预备知识

设 S为一个集合。 S上的多重集是一个函数 $m\colon S \to \mathbb{N}$，其中 $\mathbb{N} = \{0, 1, 2, \ldots\}$表示自然数集。对于 $s \in S$， $m(s)$ 表示 s 在 m中出现的次数。 S 上的所有多重集组成的集合记为 $B(S)$。我们用 $+$和 $-$ 表示两个多重集的标准加法和差。关系=，$<$，和 $\leq$ 逐元素比较多重集。我们将集合视为多重集，其中集合的每个元素在多重集中恰好出现一次。

S 上的有限序列集合记作 S^*，其中 ϵ 表示空序列，$\langle a_0, \ldots, a_n \rangle$ 表示长度为 $n+1$的序列。我们使用 $\circ$连接两个序列，并将序列 σ 的长度记作 $|\sigma|$。序列 σ的前缀是指满足 $\sigma = \sigma' \circ \sigma''$的序列 σ'，其中σ''为某个序列。当存在序列 σ'，σ'' 满足 $\sigma = \sigma' \circ \langle x \rangle \circ \sigma''$时，我们记作 $x \in \sigma$。序列 σ 在集合 Q上的投影记作 $\sigma_{|Q}$，其归纳定义如下：$\epsilon_{|Q} = \epsilon$；如果 $a \in Q$，则$(\langle a \rangle \circ \sigma')_{|Q} = \langle a \rangle \circ (\sigma'_{|Q})$；否则为 $\sigma'_{|Q|}$。我们将序列 σ 和 σ' 交错后得到的集合记作 $\sigma || \sigma'$。

带标记的佩特里网，或以下简称佩特里网，是一种用于描述复杂系统动态特性的数学模型。

定义 1. 佩特里网是一个元组 $N = (P, T, F, \mathcal{L}, \mu)$，其中 P是库所集合； T 是变迁集合，且 $P \cap T = \emptyset$，F 是流关系$F \subseteq (P \times T) \cup (T \times P)$， $\mathcal{L}$是标签集合， $\mu\colon T \to \mathcal{L}$是标记函数。

设 $N = (P, T, F, \mathcal{L}, \mu)$ 是任意但固定的佩特里网。 $P \cup T$中的元素称为节点， F 中的元素称为弧。为清晰起见，我们有时用 P_N表示网 N 的库所，用 T_N 表示变迁，对该元组的其他元素也采用类似记法。我们将节点 x 的前集定义为 ${}^\bullet x = \{x' | (x', x) \in F\}$，后集定义为 $x^\bullet = \{x' | (x, x') \in F\}$，并以标准方式将这些记法推广到节点集合。

N的路径集合$PS(N)$是满足以下条件的最小集合：$\epsilon \in PS(N)$，对于所有 $\langle x \rangle \in PS(N)$，以及对于所有 $(x_0, x_1) \in F$和有限路径 $\langle x_1, \ldots, x_n \rangle \in PS(N)$，还有 $\langle x_0, x_1, \ldots, x_n \rangle \in PS(N)$。若对所有不同的 $x, x' \in P \cup T$，都存在一条路径 $\langle x, \ldots, x' \rangle \in PS(N)$，则佩特里网 N是强连通 的。

定义 2. 当且仅当对于所有 $t \in T$:$|{}^\bullet t| \leq 1$ 和 $|t^\bullet| \leq 1$, N 是一个状态机 （*S*网）。如果恰好存在一个库所i 满足 ${}^\bullet i = \emptyset$, 称为初始库所；存在一个库所 f 满足 $f^\bullet = \emptyset$, 称为终止库所；并且所有节点 $n \in P \cup T$都位于从 i 到 f的一条路径上，则它是一个工作流网 （*WFN*）。若库所 p 满足 $|p^\bullet| > 1$ ，则称其为分叉 ；若满足 $|{}^\bullet p| > 1$, 则称其为汇合 。

若（$P_N \cup T_N$）$\cap$（$P_M \cup T_M$）$= \emptyset$, 则佩特里网 N 和 M 不相交。若存在双射 ψ：$P_N \cup T_N \to P_M \cup T_M$, 使得对所有 $p \in P_N$ ，都有 ψ（p）$\in P_M$ ；对所有 $t \in T_N$ ，都有ψ（t）$\in T_M$，且 μ_N（t）$= \mu_M$（ψ（t））；并且对所有（x, y）$\in$（$P_N \cup T_N$）$\times$（$P_N \cup T_N$），都有（x, y）$\in F_N$当且仅当（ψ（x） ψ（y））$\in F_M$，则称它们同构，记作 $N \cong M$。

佩特里网 $N =$（$P, T, F, \mathcal{L}, \mu$）的状态由其标识决定，标识表示该网各库所中的令牌分布。形式上， N 的标识是一个多重集 $m \in B$（P），其中 m（p）表示库所 $p \in P$ 中令牌的数量。若 m（p）> 0, 则库所 $p \in P$ 在标识 m 中称为有标识。若变迁 $t \in T$ 在 m 中使能，当且仅当 ${}^\bullet t \leq m$，也就是说，其前集中的所有库所都有标识。若变迁 t 已使能，则它可以从标识 m 触发，得到满足以下条件的标识 m'：对所有 $p \in P$， m'（p）$= m$（p）$-{}^\bullet t$（p）$+ t^\bullet$（p）。我们记作 $m \xrightarrow{t} m'$，表示 t 在 m 中使能，且触发 t 后得到 m'；并以预期的方式将这一概念扩展到变迁序列。

网系统是一个具有初始标识和终止标识的佩特里网；其语义由标号迁移系统给出。

定义 3. 设 $N = (P, T, F, \mathcal{L}, \mu)$ 是一个佩特里网。网系统是一个三元组$S = (N, m_0, m_f)$，其中 $m_0 \in B(P)$ 是初始标识，而 $m_f \in B(P)$ 是终止标识。

给定网系统 S，若存在某个 $\sigma \in T^*$ 使得 $m \xrightarrow{\sigma} m'$，则 S 的标识 m' 从 m 可达；我们用 $\mathcal{R}(N, m)$ 表示从 m 可达的标识集合。网系统中的库所 $p \in P_N$ 若对于所有可达标识$m \in \mathcal{R}(N, m_0)$ 都满足 $m(p) \leq 1$，则称为安全。网系统的一项重要正确性判据是它是否总能终止：这意味着系统中不存在（未知的）死锁，同时也意味着能够将行为引导至终止状态。

定义 4。 如果从所有标识 $m \in \mathcal{R}$（N、 m_0）都可以到达终止标识 m_f ，则网系统是弱终止的。

3 带标记的端口网及其镜像

开放 Petri 网（OPN）是一类研究充分的 [3,4,15,18] 佩特里网，具有开放库所和可见变迁，即包含特定的输入/输出库所和变迁集合，网可通过这些库所和变迁与其环境交互：既可以通过开放库所交换令牌，也可以在可见变迁上同步。具有开放库所的 OPN [20] 非常适合对异步通信系统建模，这类系统由网组合而成，组合时将各网共享的开放库所融合在一起，

从而表示接口。Portnet [7]是 OPN 的一个子类，通过定义与环境进行交互时允许的行为（即令牌在开放输入库所和输出库所上的交换顺序），显式地对接口行为建模。Portnet 非常适合用于描述基于组件的系统中的接口契约。提供服务的组件，其接口通过一个执行接口契约服务器端（卖方）行为的端口网，描述对环境（其他组件）的预期以及对环境承担的义务。使用另一组件所提供服务的组件充当其环境，并通过一个镜像端口网执行接口契约的客户端（买方）行为，从而定义服务的使用方式。一对端口网的兼容性不仅取决于开放库所的输入和输出标签是否匹配，还取决于对网结构施加的额外限制，以确保端口网组合具有弱终止性质。

本节介绍带标号端口网这一类，它通过标记函数以及对带标号端口网集合定义的组合操作来扩展端口网 [7]。一对端口网的兼容性有一项严格要求：它们的骨架必须同构。这使服务器端口网与其客户端之间形成镜像关系，而这种要求在实践中颇为严格，因为客户端可能只需要使用服务器端口网的一部分。我们通过端口网对之间的部分镜像关系放宽了同构要求。

开放 Petri 网（OPN）是一种佩特里网，具有一组特殊的接口库所，这些库所划分为输入库所和输出库所，用以表示该网的接口。开放 Petri 网与其环境交互时，会通过接口库所交换令牌（代表消息）。如果开放 Petri 网中的某个变迁与一个接口库所相关联，那么该库所要么是输入库所，要么是输出库所，不能两者兼是。因此，变迁表示的要么是发送，要么是接收。去除接口库所后得到的网称为其骨架。通过对骨架施加限制，可以得到开放 Petri 网的许多子类。

定义 5. 设 P、I、O 是两两不相交的集合。一个开放 Petri 网（开放 Petri 网）是一个元组$(P, I, O, T, F, \mathit{init}, \mathit{fin}\ \mathcal{L}, \mu)$，其中

$-(P \cup I \cup O, T, F, \mathcal{L}, \mu)$ 是一个佩特里网，$-$ I 是输入库所的集合；要求 ${}^\bullet I = \emptyset$，$-$ O 是输出库所的集合；要求 $O^\bullet = \emptyset$，$-$ 对于所有 $t \in T$，要么有 ${}^\bullet t \cap I = \emptyset$，要么有 $t^\bullet \cap O = \emptyset$，$-$ $\mathit{init} \subseteq P$ 是初始库所的集合，$\mathit{fin} \subseteq P$ 是终止库所的集合。

我们将 I 和 O 中的库所称为该开放 Petri 网的接口库所。开放 Petri 网中变迁相对于库所的通信方向 N是一个函数 $\lambda\colon T \to \{$发送，接收 $\tau\}$，定义如下：

$$\lambda(t) = \begin{cases} \mathit{send} & \text{if } t^\bullet \cap O \neq \emptyset \\ \mathit{receive} & \text{if } {}^\bullet t \cap I \neq \emptyset \\ \tau & \text{otherwise} \end{cases}$$

也就是说，$\lambda(t) =$ 发送 当 t 连接到输出时；而 $\lambda(t) =$ 接收 当t 连接到输入时；$\lambda(t) = \tau$ 当两种情况均不满足时。我们称 t 为

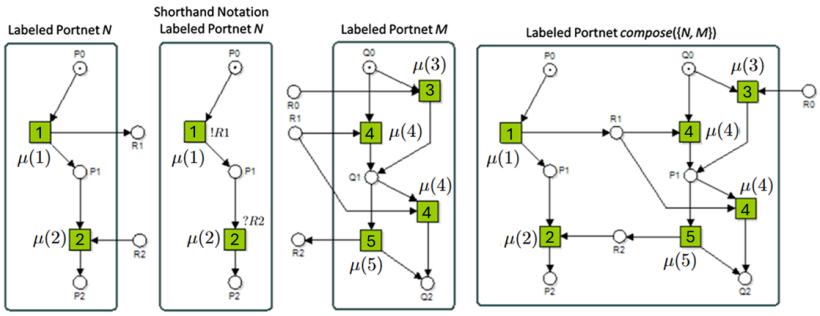


图 2：带标号端口网及其组合

通信变迁，当且仅当 $\lambda(t) \neq \tau$。 N的骨架记为 $\mathcal{S}(N)$，是一个开放 Petri 网 $\mathcal{S}(N) = (P, \emptyset, \emptyset, T, F', \text{初始化}, \text{终止}\ \mathcal{L}, \mu)$，其中 $F' = F \cap ((P \times T) \cup (T \times P))$。如果该骨架是S网，则称其为状态机开放 Petri 网（S-OPN）。如果一个开放 Petri 网 N 的骨架是工作流网，则称其为开放工作流网（OWN）。骨架为工作流网的S-OPN称为状态机开放工作流网（S-OWN）。S-OWN N的闭包记为closure(N)，是通过向 N 扩展一个新的变迁 $t \notin T$ 而得到的S-OPN，使得 $\lambda(t) = \tau$，
${}^\bullet t =$终止 和 $t^\bullet =$ 初始化。带标号端口网 是开放 Petri 网的一个子类，其结构具有额外的限制。

定义 6. 若开放 *Petri* 网 $N = (P, I, O, T, F, \text{初始化}, \text{终止}\ \mathcal{L}, \mu)$ 满足以下条件，则称其为带标号端口网：

– N 是一个 *S-OWN*，且 $|\text{初始化}| = |\text{终止}| = 1$，– 所有变迁 $t \in T$ 都满足 $|({}^\bullet t \cup t^\bullet) \cap (I \cup O)| = 1$，– 对所有 $x \in I \cup O$ 及所有变迁 $t, t' \in {}^\bullet x \cup x^\bullet$，都有 $\mu(t) = \mu(t')$，– 对所有满足 $\mu(t) = \mu(t')$ 的变迁 $t, t' \in T$，都有 $({}^\bullet t \cup t^\bullet) \cap (I \cup O) = ({}^\bullet t' \cup t'^\bullet) \cap (I \cup O)$，

开放 *Petri* 网 N 的网系统 (N, m_0, m_f) 以 $m_0 = \mathit{init}$ 为初始标识，以 $m_f = \mathit{fin}$ 为终止标识，其中集合 *init* 和 *fin* 被解释为多重集。

Portnet 如 [7] 中所定义，是带标记端口网的实例，其标记函数 μ 满足 $\mu(t) = t$。也就是说，端口网中的每个变迁都有其独一无二的名称。为了强调端口网与带标记端口网之间的区别，我们有时将前者称为普通端口网。

引理 1. 设 N 是一个带标号端口网，且 $t, t' \in T$ 满足 $\mu(t) = \mu(t')$。则 $\lambda(t) = \lambda(t')$。

证明。 任取 t， $t' \in T$ 满足 $\mu(t) = \mu(t')$。首先注意，根据对带标号端口网的第二项要求，恰好存在一个 $x \in I \cup O$ 使得 $x \in {}^\bullet t \cup t^\bullet$。假设 $x \in I$；情形 $x \in O$ 完全对偶。由于根据开放 Petri 网的第二个条件， ${}^\bullet I = \emptyset$，我们必然有 $x \in {}^\bullet t$，因此根据定义， $\lambda(t) =$ 接收。由于根据对带标号端口网的第四项要求，我们发现 $x \in {}^\bullet t' \cup t'^\bullet$，因此要么 $x \in {}^\bullet t'$ ，要么 $x \in t'^\bullet$。假设 $x \in t'^\bullet$。

于是 $t' \in {}^\bullet I$，这与开放 Petri 网的条件 ${}^\bullet I = \emptyset$ 矛盾。因此 $x \in {}^\bullet t'$，从而λ(*t'*) = 接收 = λ(*t*)。□

由于带标号端口网的骨架是一个状态机工作流网，根据定理 15 [10], ，它是弱终止的，如下述推论所述。

推论 1。带标号端口网的骨架弱终止。

图 2 展示了一个带标号端口网示例。如图所示，将使用简写符号？（接收）和！（发送）以保持图形清晰；必要时，会在其前加上所连接接口库所的名称作为前缀。在变迁上的标签旁，使用数字作为其标识符。根据上下文，我们可以省略其中之一。

如果两个开放 Petri 网共享的唯一节点是接口库所，并且当它们共享非空的接口库所集合时，其中一个的输出是另一个的输入的子集，或者它们的输入和输出互不相交，那么这两个开放 Petri 网是可组合的。一组两两可组合的OPN的组合，几乎是其相应元组的两两并集，区别在于 OPN 之间共享的接口库所会成为该组合的内部库所。图 2 展示了两个可组合的带标记的端口网 N 和 M的组合。

定义 7. 两个开放 *Petri* 网 N 和 M 是可组合的 当且仅当

$-(P_N \cup I_N \cup O_N \cup T_N) \cap (P_M \cup I_M \cup O_M \cup T_M) = ((I_N \cup O_N) \cap (I_M \cup O_M))$，$-$若 $(I_N \cap O_M) \cup (I_M \cap O_N) \neq \emptyset$，则还有：

- $O_N \subseteq I_M$ 和 $O_M \subseteq I_N$，并且
- $I_N \cap I_M = O_N \cap O_M = \emptyset$。

一组非空且两两可组合的 *OPN* S 的组合是一个 *OPN compose*$(S) = (P, I, O, T, F,$ 初始化*, fin* $\mathcal{L}, \mu)$，其中

- $P = (\bigcup_{N \in S} P_N) \cup ((\bigcup_{N \in S} I_N) \cap (\bigcup_{N \in S} O_N))$,
- $I = (\bigcup_{N \in S} I_N) \setminus (\bigcup_{N \in S} O_N)$, $O = (\bigcup_{N \in S} O_N) \setminus (\bigcup_{N \in S} I_N)$,
- $T = (\bigcup_{N \in S} T_N)$, $F = (\bigcup_{N \in S} F_N)$,
- 初始化 $= (\bigcup_{N \in S}$ 初始化$_N)$，终止 $= (\bigcup_{N \in S}$ 终止$_N)$,
- $\mathcal{L} = (\bigcup_{N \in S} \mathcal{L}_N)$ 和 $\mu = (\bigcup_{N \in S} \mu_N)$。

我们注意到，由于 OPN 的共享库所在复合时会变成内部库所，因此对于两两可组合的 OPN 集合 S_1 和 S_2 ，一般而言，我们有 复合$(S_1 \cup S_2)$ = 复合$(S_1 \cup$复合$(S_2))$。

部分镜像关系放宽了一对端口网 [7] 的骨架上的同构要求，允许删除一些方向为发送的变迁，从而让设计者可以只使用另一带标号端口网中的相关部分。

定义 8。带标号端口网 N 的部分镜像是一个二元组（M、ϕ），其中 M是一个带标号端口网，且 ϕ： $M \to N$ 是满足以下条件的单射映射：

- $\phi(P_M) \subseteq P_N$, $\phi(T_M) \subseteq T_N$, $\phi(I_M) \subseteq O_N$ *and* $\phi(O_M) \subseteq I_N$,

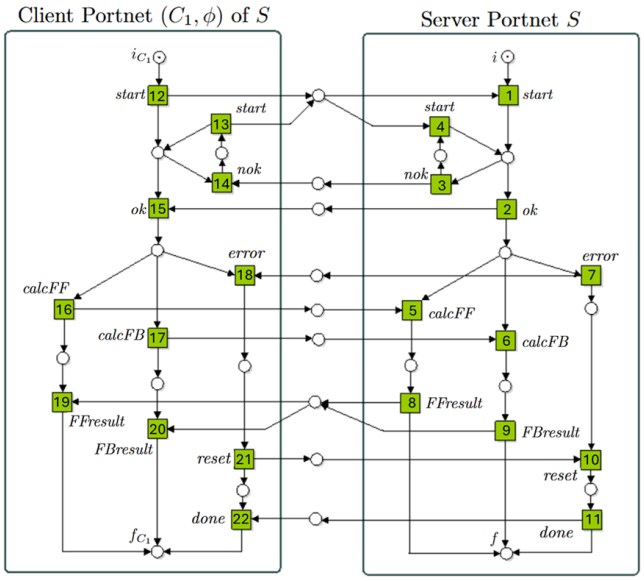


图3：带标号端口网及其镜像组合中的死锁

– F_M满足对所有（x、 y） $\in F_M$:

●若 $x \notin I_M$且 $y \notin O_M$，则 $(\phi(x)\ \phi(y)) \in F_N$，

●若 $x \in I_M$或 $y \in O_M$，则 $(\phi(y)\ \phi(x)) \in F_N$，– 对所有 $t \in T_M$，都有 $\mu_M(t) = \mu_N(\phi(t))$，– 对所有库所 $p \in P_M$和所有变迁 $t \in \phi(p)^\bullet$在 N中，若 $\lambda(t) =$ 发送，则存在某个 $t' \in p^\bullet$ 使得 $\phi(t') = t$。

若 ϕ 是双射，我们称 (M, ϕ) 为一个镜像标记端口网。带标号端口网 M 称为服务器端口网 N的客户端口网。

部分镜像关系本身并不能保证带标号端口网与其（部分）镜像的组合具有弱终止性。图3展示了一个关于 ϕ的此类组合示例，其中一个死锁可以通过发射变迁序列 $\langle 12,1,2,15,7,16\rangle$到达。

4 良构标记端口网

在 [7]，中，我们引入了普通端口网对之间的一种兼容性概念，该概念可以保证它们组合后的弱终止性。这一兼容性概念要求普通端口网的骨架满足选择和支路性质；为完整起见，我们将在下文重复介绍这些性质。这些是普通端口网的结构性质，可以高效地进行检查。

“选择性质”不允许库所后集中的变迁在输入与输出之间产生竞争：这些变迁要么全部执行

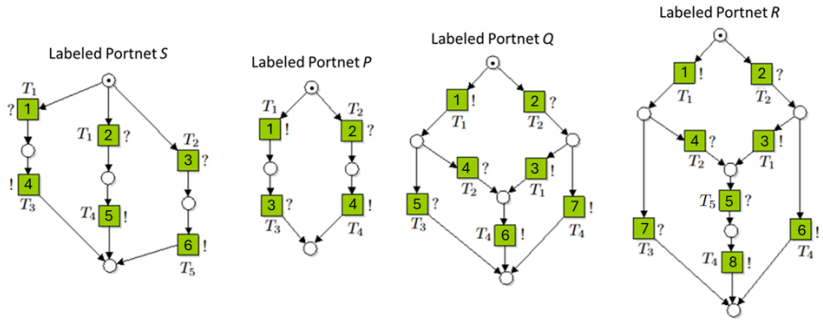


图 4：端口网性质

发送，要么全部执行接收，不能两者混合。因此，无法对客户端与服务器端口网之间的竞争进行建模。“支路性质”可避免客户端与服务器端口网陷入混淆，沿不同路径前进。该性质要求从分叉（后集大于 1 的库所，包括初始库所）到汇合（前集大于 1 的库所，包括终止库所）的每条路径上，至少有两个通信方向不同的变迁，从而避免混淆。

定义 *9*（选择与支路性质）。若端口网 N 满足以下条件，则称其满足选择性质：对于所有 $p \in P_N$，对于所有 $t, t' \in p^\bullet$，均有 $\lambda(t) = \lambda(t')$。若 $\forall \beta = \langle p_1, t_1, \ldots, t_{n-1}, p_n \rangle \in PS(N)$：$(|p_1{}^\bullet| > 1 \vee p_1 = \text{初始化}_N) \wedge (|{}^\bullet p_n| > 1 \vee p_n = fin_N)$：$\exists t, t' \in \beta$：$\lambda(t) \neq \tau \wedge \lambda(t') \neq \tau \wedge \lambda(t) \neq \lambda(t')$，则称其满足支路性质。

本章节通过引入可观察的选择、菱形性质 和循环 性质，对这些性质进行了放宽。此外，我们在更一般的带标记端口网设置下考察这些性质。因此，满足这些性质的带标记端口网涵盖了种类更丰富的交互协议，这些协议在实践中十分常见。

若一个带标号端口网满足 可观测选择 性质，则某个库所的后集中不同的变迁具有不同的标记。这样一来，仅根据观测到的标记，就能准确判断带标号端口网中的哪个非接口位置会在发射一个变迁序列后变为有标识。例如，图 4 中的端口网 S 不满足该性质，而其他端口网均满足。

定义 10（可观测选择）。若一个带标号端口网满足可观测选择性质，则对所有 $p \in P$，以及所有不同的 t、$t' \in p^\bullet$，均有 $\mu(t) \neq \mu(t')$。

下文所述的菱形性质放宽了前述的选择性质，允许客户端和服务器之间发生竞争，从而从一个分叉库所开始向彼此发送消息。直观而言，无论竞争以何种方式解决，只要落败一方的交互处理选项仍然存在，并且之后处理该交互仍有可能使双方对彼此状态的看法保持一致，这种竞争就是良性的。下文将对这一直觉作形式化表述。

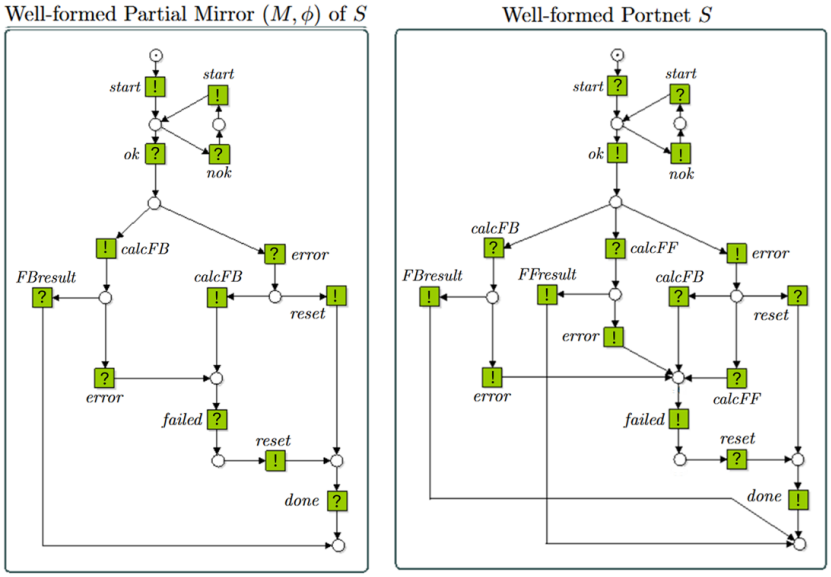


图 5：良构端口网及其良构部分镜像端口网

定义 11（菱形性质）。如果带标号端口网的所有库所 $p \in P$ 以及所有 t、$t' \in p^\bullet$ 满足 $\lambda(t) \neq \lambda(t')$，则称其满足菱形性质

$$\forall q \in t^\bullet, q' \in t'^\bullet : \exists u \in q^\bullet, u' \in q'^\bullet : u^\bullet \cap u'^\bullet \neq \emptyset \wedge \mu(t) = \mu(u') \wedge \mu(t') = \mu(u)$$

性质 *1.* 如果带标号端口网满足选择性质，则它也满足菱形性质。

图 4 所示的带标号端口网 S 满足选择性质，而其他带标号端口网不满足该性质。带标号端口网 P 不满足菱形性质，而带标号端口网 Q 和 R 满足该性质。

定义 12. 对于给定的带标号端口网 N（其库所为 $p \in P$、变迁为 t、标记为 $l \in \mathcal{L}$），我们将路径集(p, t, l) 定义如下：

$$\{\langle p \rangle \circ \pi \in PS(N) \mid \exists \sigma \in T^*, t'' \in T : \pi_{|T} = \langle t \rangle \circ \sigma \circ \langle t'' \rangle \wedge \mu(t'') = l \wedge \forall u \in T : u \in \sigma \Rightarrow \mu(u) \neq l\}$$

下面形式化定义的循环性质要求：任何以重新解决相互竞争的接收（分别地，发送）交互为起点、并通过处理其中一个交互来解决问题的路径，也必须处理另一个交互，但不能早于一次发送（分别地，接收）交互发生。这样做是为了避免客户端和服务器产生混淆，导致双方走上不同的路径。例如，考虑端口网 Q（图 4）中某个分叉的后集中发送变迁 3 和 7。假设 Q触发了由发送 变迁组成的序列 $\langle 3, 6 \rangle$。那么，它的部分镜像（X，ϕ）可以选择触发其接收 变迁 t，其中 ϕ（t）$= 7$（因为 6 和 7 具有相同的

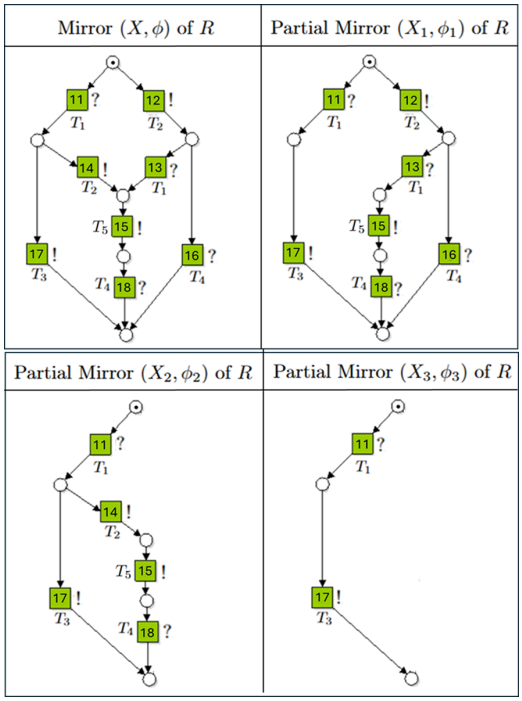


图 6：图 4 中良构端口网 R 的部分镜像

标记 T_4，它们连接到同一个接口库所），从而导致变迁 3 的接口库所中留下令牌。端口网 R 通过强制与接收 变迁 5 同步来解决这一问题。

定义 13（循环性质）。如果一个带标号端口网中的所有库所 $p \in P$ 都满足：对于任意两个不同的变迁 t、$t' \in p^\bullet$，若 $\lambda(t) = \lambda(t')$，且对于所有路径 $\sigma \in$ 路径集$(p, t, \mu(t'))$，都存在一个变迁 $t'' \in \sigma$，使得 $\lambda(t'') \neq \lambda(t)$，则该带标号端口网满足循环性质。

定义 14（良构端口网）。如果一个带标号端口网满足可观测选择、菱形性质和循环性质，则称其为良构。

图5所示的端口网 S 是对图3所示端口网 S 的修改；这些修改确保该端口网是良构的。其良构的部分镜像 M 去掉了标记为 calcFF 的变迁，因此也去掉了冲突变迁 FFResult 和 error，以确保 M 的骨架仍为工作流网。我们指出，带标号端口网的良构性不会在部分镜像后得以保留。为说明这一点，考虑图4所示的良构端口网 R 及其在图6中的部分镜像，其中 ϕ(11) $= \phi_1$(11) $= \phi_2$(11) $= \phi_3$(11) $= 1$, ϕ(12) $= \phi_1$(12) $= 2$, ϕ(13) $= \phi_1$(13) $= 3$, ϕ(14) $= \phi_2$(14) $= 4$,ϕ(15) $= \phi_1$(15) $= \phi_2$(15) $= 5$, ϕ(16) $= \phi_1$(16) $= 6$, ϕ(17) $= \phi_1$(17) $= \phi_2$(17) $= \phi_3$(17) $= 7$, 以及 ϕ(18) $= \phi_1$(18) $= \phi_2$(18) $= 8$。部分镜像 (X_1, ϕ_1)

不是良构的，因为它违反了菱形性质。然而，另外两个部分镜像 (X_2, ϕ_2) 和 (X_3, ϕ_3) 是良构的。镜像标记端口网 (X, ϕ) 也是良构的。

5 标记端口网组合的弱终止

如图3中的示例所示，带标号端口网与其部分镜像的组合不一定会得到一个弱终止的开放 Petri 网。本章节将证明，当两个带标记的端口网 N 和 M 都是良构的，且 M 是 N的部分镜像时，它们的组合保证弱终止。本章节的主要定理如下:

定理 1. 设 N 和 M 是良构标记端口网，且 (M, ϕ) 是 N的部分镜像。设 $S =$ 复合$(\{N, M\})$， $i_S = i_N + i_M$是 S 的初始标识，且 $f_S = f_N + f_M$ 是 S的终止标识。那么，网系统 (S, i_S, f_S) 弱终止。

在本章节的其余部分中，我们将逐步阐述该定理的正确性论证，最后给出其证明。

固定一个良构的带标号端口网 N——以下称为服务器——以及一个 N 的良构部分镜像 (M, ϕ)——以下称为客户端。假设在它们交互的某个时刻，服务器希望执行一个发送变迁。由于通信的异步性，客户端可能还在追赶之前已执行的发送变迁。不过，由于菱形性质，服务器究竟何时触发其发送变迁并不重要，因为这对系统整体状态的影响相同。下面的引理将这一现象形式化。

引理 2. 设 N 和 M 是良构的带标号端口网，且 (M, ϕ) 是 N 的部分镜像。令 $S =$*compose*$(\{N, M\})$，且 $i_S = i_N + i_M$ 是 S 的初始标识， $f_S = f_N + f_M$ 是 S 的终止标识。对于任意标识 $m, m', m_0, \ldots, m_n \in \mathcal{R}(S, i_S)$，以及 N 的所有变迁 $t, t_1, \ldots, t_n$，使得 $m = m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_n} m_n$，其中每个 $\lambda(t_i) = receive$，且 $m \xrightarrow{t} m'$ 满足 $\lambda(t) = send$，存在标识 $m'_0, \ldots, m'_n \in \mathcal{R}(S, i_S)$ 和 N 的变迁 $\bar{t}, t'_1, \ldots, t'_n$，使得 $m' = m'_0 \xrightarrow{t'_1} \ldots \xrightarrow{t'_n} m'_n$ 和 $m_n \xrightarrow{\bar{t}} m'_n$，并且

1. $\mu(t_i) = \mu(t'_i)$ 和 $\lambda(t_i) = \lambda(t'_i)$;
2. $\mu(t) = \mu(\bar{t})$ 和 $\lambda(t) = \lambda(\bar{t})$。

证明。 我们对端口网 N 中接收变迁序列 $t_1 \ldots t_n$ 的长度进行归纳（这些变迁来自 M 的情况与此类似）。

– 基例， $n = 0$。此时不存在接收变迁，因此该陈述自然成立。–归纳步骤。假设该陈述对序列 $m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_n} m_n$ 成立。假设 $m_n \xrightarrow{t_{n+1}} m_{n+1}$ 对于某个由以下内容触发的接收变迁 t_{n+1},

端口网 N 和 $m_0 \xrightarrow{t} m_0'$，满足 $\lambda(t) = send$。根据归纳假设，我们知道 $m_0' \xrightarrow{t_1'} m_1' \dots \xrightarrow{t_n'} m_n'$，满足 $\mu(t_i) = \mu(t_i')$，并且 $\lambda(t_i) = \lambda(t_i')$。此外，$m_n \xrightarrow{\bar{t}} m_n'$ 对于满足 $\mu(t) = \mu(\bar{t})$ 以及 $\lambda(\bar{t}) = send$ 的某个 $\bar{t}$。由于 $m_n \xrightarrow{t_{n+1}} m_{n+1}$ 和 $m_n \xrightarrow{\bar{t}} m_n'$，我们得到 ${}^\bullet t_{n+1} \cap {}^\bullet \bar{t} \neq \emptyset$。根据菱形性质，我们随后发现必定存在某个 $\tilde{t}$ 和 m_{n+1}'，使得 $m_{n+1} \xrightarrow{\tilde{t}} m_{n+1}'$，并且存在某个 t_{n+1}'，使得 $m_n' \xrightarrow{t_{n+1}'} m_{n+1}'$ 和 $\mu(\tilde{t}) = \mu(\bar{t})$ 以及 $\mu(t_{n+1}') = \mu(t_{n+1})$，从而完成证明。□

与交换引理（见 [9]）类似，我们证明，只要某个使能变迁与该可执行发射序列中各变迁的前集不共享任何库所，就可以在发射序列的任意位置触发该变迁。

引理 3. 设 (N, m_0, m_f) 是一个网系统，其中 N 是一个带标号开放 *Petri* 网。假设 $t \in T$ 和 $\sigma \in T^*$ 满足：对于所有 $t' \in \sigma$，${}^\bullet t'$ 与 ${}^\bullet t \cup t^\bullet$ 不相交。那么，对于每个满足 $m \xrightarrow{t}$ 和 $m \xrightarrow{\sigma}$ 的 $m \in \mathcal{R}(N, m_0, m_f)$，都存在某个 m'，使得对于每个 $\sigma' \in (\langle t \rangle || \sigma)$，都有 $m \xrightarrow{\sigma'} m'$。

证明。 根据标识方程引理 [9], , $(\langle t \rangle || \sigma)$ 中所有可能的序列都会导向相同的结果标识（前提是各变迁的出现次数相同）。因此，我们只需证明 $(\langle t \rangle || \sigma)$ 中任意序列都是可执行发射序列。设 α' 和 α'' 满足 $\alpha = \alpha' \circ \langle t \rangle \circ \alpha''$，即 $\sigma = \alpha' \circ \alpha''$ 和 $\alpha \in (\langle t \rangle || \sigma)$。因此，对于某个 $\tilde{m}$，$\tilde{m}' \in \mathcal{R}(N, m)$，有 $m \xrightarrow{\alpha'} \tilde{m} \xrightarrow{\alpha''} \tilde{m}'$。由于 ${}^\bullet t$ 和 ${}^\bullet \alpha'$ 不相交，变迁 t 在 $\tilde{m}$ 中使能。因此，存在一个标识 $\tilde{m}'' \in \mathcal{R}(N, m)$，满足 $\tilde{m} \xrightarrow{t} \tilde{m}''$。由于 $t^\bullet$ 和 ${}^\bullet \alpha''$ 不相交，$\tilde{m}'' \xrightarrow{\alpha''} m'$，由此引理得证。□

带标号端口网的一个接收变迁与其部分镜像的一个接收变迁不共享接口库所。因此，在良构带标号端口网与其良构部分镜像的组合中，前者的接收变迁序列可以与后者交错执行，并得到相同的标识。

引理 4. 设 N 和 M 是良构带标号端口网，且 (M, ϕ) 是 N 的部分镜像。设 $S =$ 复合$(\{N, M\})$，$i_S = i_N + i_M$ 是 S 的初始标识，$f_S = f_N + f_M$ 是 S 的终止标识。对于任意可达标识 $m_0, \dots, m_n$，以及所有变迁 $t_1, \dots, t_n \in T_N$ 和 $u_1, \dots, u_k \in T_M$，使得 $m_0 \xrightarrow{t_1} \dots \xrightarrow{t_n} m_n \xrightarrow{u_1} \dots \xrightarrow{u_k} m_{n+k}$ 和 $\lambda(t_i) = \lambda(u_j) = receive$ 对所有相关的 i, j 成立，存在标识 $m_0', \dots, m_{n+k}'$，使得 $m_0 = m_0' \xrightarrow{u_1} \dots \xrightarrow{u_k} m_k' \xrightarrow{t_1} \dots \xrightarrow{t_n} m_{n+k}' = m_{n+k}$。

证明。 由引理 3 可知，结论成立，因为良构带标号端口网的接收变迁与其部分镜像的接收变迁不会共享接口库所。□

从良构服务器端口网与其良构部分镜像客户端的组合中的一个可达标识出发，发射服务器的一列使能接收变迁不会使客户端的一个使能发送变迁失能，因为该变迁不会从接口库所中消耗标识。

引理 5. 设 N 和 M 是良构标记端口网，且 (M, ϕ) 是 N的部分镜像。令 $S =$复合$(\{N, M\})$，且 $i_S = i_N + i_M$是 S 的初始标识， $f_S = f_N + f_M$是 S的终止标识。对于任意可达标识$m_0, \ldots, m_n, m'_0$，以及所有满足 $\lambda(t_i) = receive$的变迁 $t_1, \ldots, t_n \in T_N$，和满足 $\lambda(u) = send$ 的$u \in T_M$，使得 $m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_n} m_n$和 $m_0 \xrightarrow{u} m'_0$，存在标识 $m'_1, \ldots, m'_n$，使得 $m'_0 \xrightarrow{t_1} \ldots \xrightarrow{t_n} m'_n$和 $m_n \xrightarrow{u} m'_n$。

证明。由于发送变迁 u与任何接收变迁 $t_1 \ldots t_n$的前集都不重合，因此变迁 u 在标识 m_n中仍然使能。□

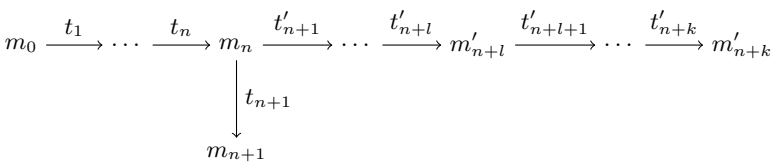
基于前述引理，下一个命题表明，对于良构服务器端口网与其良构部分镜像客户端的组合中任意可达的标识，都存在一个接收变迁序列，可以先在服务器中执行，再在客户端中执行，从而到达一个标识，使客户端中的某个库所及其镜像均有标识，而所有其他库所均为空。通俗地说，这意味着服务器和客户端只需处理接收变迁，就能彼此追上，并使双方对彼此状态的认知保持一致。

命题 1。设 N 和 M 为良构标记端口网，且 (M, ϕ)是 N的部分镜像。令 $S =$复合$(\{N, M\})$， $i_S = i_N + i_M$为 S 的初始标识， $f_S = f_N + f_M$ 为 S 的终止标识。对于任意标识 $m_0, \ldots, m_n$以及满足 $i_S = m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_n} m_n$的所有变迁 $t_1, \ldots, t_n$，存在标识 $m_{n+1}, \ldots, m_{n+k}$和变迁 $t_{n+1}, \ldots, t_{n+k}$，使得 $m_n \xrightarrow{t_{n+1}} \ldots \xrightarrow{t_{n+k}} m_{n+k}$并且：

1. 对所有 $i \geq n+1$，都有 $\lambda(t_i) = receive$；*2.* $t_{n+1} \ldots t_{n+k} \in T_N^* T_M^*$；即 这些变迁先在 N中执行，然后在 M中执行；*3.* 存在一个库所 $p \in P_M$，使得 $m_{n+k}(p) = m_{n+k}(\phi(p)) = 1$，且对于所有 $p' \notin P_N \cup P_M$， $m_{n+k}(p') = 0$；即 所有接口库所均为空，并且在 M 中的库所 p 及其在 N中的镜像 $\phi(p)$ 中都各有一个令牌。

证明。我们对路径 $m_0 \ldots m_n$的长度进行归纳。

– 基例， $n = 0$。此时只有一个标识，即*viz.* m_0，且没有变迁触发。因此，空标识序列（取 $k = 0$）自然满足所需的三个条件。–归纳步骤。考虑标识 $m_0, \ldots, m_{n+1}$ 以及变迁 $t_1, \ldots, t_{n+1}$，使得 $i_S = m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_{n+1}} m_{n+1}$；并假设对于 $i_S = m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_n} m_n$，存在标识 $m'_{n+1}, \ldots, m'_{n+k}$以及变迁 $t'_{n+1}, \ldots, t'_{n+k}$，使得 $m_n \xrightarrow{t'_{n+1}} \ldots \xrightarrow{t'_{n+k}} m'_{n+k}$满足所需的三个性质。那么，根据归纳假设，情况如下：



对于某个（非接口）库所 $p \in P_M$，有 $\lambda(t'_{n+1}) = \cdots = \lambda(t'_{n+k}) = receive$，且对于所有 $p \notin P_N \cup P_M$，$m'_{n+k}(p) = m'_{n+k}(\phi(p)) = 1$。此外，我们有 $t'_{n+1}, \ldots, t'_{n+l} \in T_N$ 以及 $t'_{n+l+1}, \ldots, t'_{n+k} \in T_M$。

根据 t_{n+1} 的通信方向，我们区分两种情况。

- 情况 $\lambda(t_{n+1}) = send$。我们进一步根据端口网的发射情况 t_{n+1} 和 t'_{n+1}进行区分。
  - t_{n+1}和 t'_{n+1}都属于 T_N。根据引理 2，存在一条路径 $m_{n+1} \xrightarrow{t''_{n+1}} \ldots \xrightarrow{t''_{n+l}} m_{n+l+1}$，使得对于所有 $1 \le i \le l$，$\mu(t''_{n+i}) = \mu(t'_{n+i})$，并且 $\lambda(t''_{n+i}) = \lambda(t''_{n+i})$；此外，对于满足 $\mu(\tilde{t}_{n+1}) = \mu(t_{n+1})$ 和 $\lambda(\tilde{t}_{n+1}) = send$的某个$\tilde{t}_{n+1}$，$m'_{n+l} \xrightarrow{t_{n+1}} m_{n+l+1}$：

$$\begin{array}{ccccccccccccc} m_0 & \xrightarrow{t_1} & \cdots & \xrightarrow{t_n} & m_n & \xrightarrow{t'_{n+1}} & \cdots & \xrightarrow{t'_{n+l}} & m'_{n+l} & \xrightarrow{t'_{n+l+1}} & \cdots & \xrightarrow{t'_{n+k}} & m'_{n+k} \\ & & & & \downarrow{\scriptstyle t_{n+1}} & & & & \vdots{\scriptstyle \tilde{t}_{n+1}} & & & & \\ & & & & m_{n+1} & \xrightarrow{t''_{n+1}} & \cdots & \xrightarrow{t''_{n+l}} & m_{n+l+1} & & & & \end{array}$$

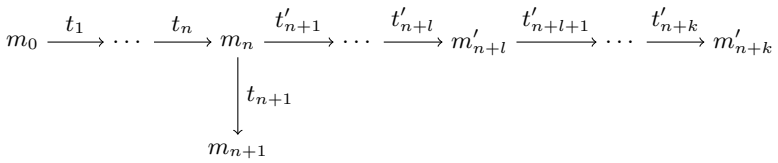
    根据引理 5，还可以进一步得出，我们能够扩展该执行：

$$\begin{array}{ccccccccccccc} m_0 & \xrightarrow{t_1} & \cdots & \xrightarrow{t_n} & m_n & \xrightarrow{t'_{n+1}} & \cdots & \xrightarrow{t'_{n+l}} & m'_{n+l} & \xrightarrow{t'_{n+l+1}} & \cdots & \xrightarrow{t'_{n+k}} & m'_{n+k} \\ & & & & \downarrow{\scriptstyle t_{n+1}} & & & & \vdots{\scriptstyle \tilde{t}_{n+1}} & & & & \vdots{\scriptstyle \tilde{t}_{n+1}} \\ & & & & m_{n+1} & \xrightarrow{t''_{n+1}} & \cdots & \xrightarrow{t''_{n+l}} & m_{n+l+1} & \xrightarrow{t'_{n+l+1}} & \cdots & \xrightarrow{t'_{n+k}} & m_{n+k+1} \end{array}$$

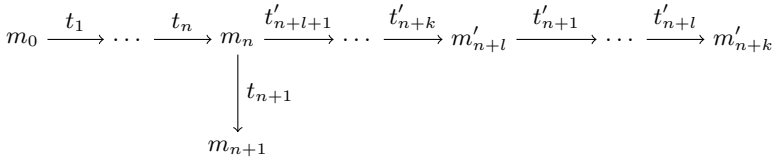
    现在只需论证存在一个变迁 t'_{n+k+1}，使得$\mu(t'_{n+k+1}) = \mu(t_{n+1})$，并且 $\lambda(t'_{n+k+1}) = receive$ 已使能。根据归纳假设，我们可以找到某个库所 $p \in P_M$和镜像 $\phi(p) \in P_N$，使得 $m'_{n+k}(p) = m'_{n+k}(\phi(p)) = 1$。令 $p \in P_M$ 和 $\bar{p} = \phi(p) \in P_N$ 为这样的对象。注意，在顶部执行序列中，我们发现 N 不再在标识 m'_{n+l}中执行变迁，因此对于所有 $q \in P_N$，$m'_{n+l}(q) = m'_{n+k}(q)$。

    由于 $m'_{n+l}(\bar{p}) = 1$，N 是一个 S-OWN，且 $m'_{n+l} \xrightarrow{t_{n+1}}$，我们必有 $\bar{p} \in {}^\bullet\tilde{t}_{n+1}$。令 $p' \in \tilde{t}^\bullet_{n+1} \cap P_N$，并令 $\tilde{t}_{n+1} \in T_M$ 满足 $\phi(\tilde{t}_{n+1}) = \tilde{t}_{n+1}$。注意，由于 $\lambda(\tilde{t}_{n+1}) =$发送，$(\phi(p), \tilde{t}_{n+1}) \in F_N$，且 M 是部分镜像（根据定义 8），这个 $\tilde{t}_{n+1} \in T_M$ 存在。于是 $\lambda(\tilde{t}_{n+1}) =$ 接收。由于 $\bar{p} \in {}^\bullet\tilde{t}_{n+1}$ 和 $\phi(\tilde{t}_{n+1}) = \tilde{t}_{n+1}$，我们有 $p \in {}^\bullet\tilde{\tilde{t}}_{n+1}$ 和 ${}^\bullet\tilde{\tilde{t}}_{n+1} \subseteq \tilde{t}^\bullet_{n+1} \cup \{p\}$，我们可以执行 $m_{n+k+1} \xrightarrow{t_{n+1}} m_{n+k+2}$，使得 $m_{n+k+2}(p') = m_{n+k+2}(\bar{p}') = 1$，对于 $\bar{p}' \in \tilde{\tilde{t}}^\bullet_{n+1} \cap P_M$（注意：$\bar{p}'$ 存在，因为镜像的骨架是工作流网——定义 8），使得 $\phi(\bar{p}') = p'$，并且对于所有 $i \notin \{p', \bar{p}'\}$，$m_{n+k+2}(i) = 0$。
  - t_{n+1}属于 T_N，但 t'_{n+1}属于 T_M。这种情况是前一种情况的简化，因为根据归纳假设，我们有 t'_{n+1}，……，$t'_{n+k} \in T_M$。

* t_{n+1}属于 T_M，但 t'_{n+1}属于 T_N。根据引理 4，我们可以进行如下变换：



to:



但这样一来，论证又归结为第一种情况。随后得到的变迁序列还可以再次使用引理 4 进行重新排列，从而得到 $T_N^* T_M^*$ 中的变迁序列。

* t_{n+1} 和 t'_{n+1} 都属于 T_M。这种情况同样是前一种情况的简化。

- 情形 $\lambda(t_{n+1}) = receive$。我们再次根据端口网触发 t_{n+1} 和 t'_{n+1}来分类讨论。

  * t_{n+1} 和 t'_{n+1} 都属于 T_N。我们证明 $t_{n+1} = t'_{n+1}$，由此即可立即得到所需结论。为导出矛盾，假设 $t_{n+1} \neq t'_{n+1}$。由于 t_{n+1} 和 t'_{n+1} 均已使能，且 N 和 M 是 S 网，因此必有 ${}^\bullet t_{n+1} \cap {}^\bullet t'_{n+1} \neq \emptyset$。此外，与 t_{n+1} 和 t'_{n+1} 相连的接口库所都至少包含一个令牌。根据归纳假设，在标识 m'_{n+k} 中，不存在包含令牌的接口库所。因此，变迁 $t \in \{t'_{n+2}, \ldots, t'_{n+k}\}$ 中至少有一个必须满足 $\mu(t) = \mu(t_{n+1})$。但根据循环性质，这不可能成立，因为在触发任何带有标记 $\mu(t_{n+1})$ 的变迁之前，必须先触发变迁 t'，且该变迁满足 $\lambda(t') = send$。矛盾，因此 $t_{n+1} = t'_{n+1}$。

  * t_{n+1}属于 T_N，但 t'_{n+1}属于 T_M。这不可能成立，因为这与以下假设矛盾：变迁序列 $t'_{n+1} \ldots t'_{n+k}$ 仅涉及 M 发射，却能清空所有接口库所。然而， M 的接收变迁无法清空与 t_{n+1} 相连的接口库所。

  * t_{n+1}属于 T_M，但 t'_{n+1}属于 T_N。利用引理 4，我们可以交换变迁序列 $t'_{n+1} \ldots t'_{n+l}$ 和 $t'_{n+l+1} \ldots t'_{n+k}$。按照第一种情况的推理，我们得到 $t'_{n+l+1} = t_{n+1}$。再应用一次引理 4，即可得出所需结论。

  * t_{n+1} 和 t'_{n+1} 都属于 T_M。这种情况是前一种情况的简化。□

从良构服务器端口网与其良构部分镜像客户端的组合中的一个可达标识出发，其中客户端中的一个库所及其镜像

有标识，且所有其他库所均为空，可以触发发送变迁与镜像接收变迁交替组成的序列，从而到达终止标识。证明依赖于端口网骨架的弱终止性（推论 1）。

命题 *2.* 设 N 和 M 是良构标记端口网，且 (M, ϕ) 是 N的部分镜像。设 $S =$ 复合$(\{N, M\})$，$i_S = i_N + i_M$是 S 的初始标识，$f_S = f_N + f_M$ 是 S的终止标识。假设 M的骨架是弱终止的。设 $m \in \mathcal{R}(S, i_S, f_S)$，使得对于某个 $p \in P_M$，$m(p) = m(\phi(p)) = 1$，并且对于所有 $q \notin \{p, \phi(p)\}$，$m(q) = 0$。那么标识 f_S 可从标识 m到达。

证明。我们首先引入用于排列两个变迁的简写符号。设t，t'是两个变迁。我们用 $t \langle\rangle t'$表示：若 $\lambda(t) = send$，则 tt'；否则为 $t't$。接下来我们重点证明以下性质：

$\forall k \in \mathbb{N}$和 $\forall m \in \mathcal{R}(S, i_S)$，使得对于某个 $p \in P_M$，$m(p) = m(\phi(p)) = 1$且 $m(q) = 0$ 对所有 $q \notin \{p, \phi(p)\}$，若 $t_1 \dots t_k$ 是骨架中将令牌从 p 移至 fin_M 的最短发射序列，且 M 的长度为 k，那么在 S中从标识 m 发射 $(t_1 \langle\rangle \phi(t_1)) \dots (t_k \langle\rangle \phi(t_k))$，会将 p和 $\phi(p)$ 中的令牌分别移至 fin_M 和 fin_N，所得标识为 f_S。

我们对 k 使用归纳法来证明这一性质。

– 基例。设可达标识 $m \in \mathcal{R}(S, i_S)$ 满足：对于某个$p \in P_M$，$m(p) = m(\phi(p)) = 1$ 且 $m(q) = 0$ 对所有 $q \notin \{p, \phi(p)\}$。设 $t_1 \dots t_k$ 是骨架中将令牌从库所 p 移至 fin_M 的最短发射序列，且其长度为 $k = 0$。由此可知，$p = fin_M$，且 $\phi(p) = fin_N$。由于 S 中没有其他令牌，我们得出 $m = f_S$，因此命题成立。

– 归纳情形。作为归纳假设，我们假定，对于所有可达标识 $m' \in \mathcal{R}(S, i_S)$，若存在某个 $p' \in P_M$, $m'(p') = m'(\phi(p')) = 1$且 $m'(q') = 0$对所有 $q' \notin \{p', \phi(p')\}$均成立，并且 $t_1 \dots t_k$，则在 M 的骨架中，将令牌从库所 p' 移至 终点$_M$ 的最短发射序列长度为 k；那么，在 S 中发射 $(t_1 \langle\rangle \phi(t_1)) \dots (t_k \langle\rangle \phi(t_k))$ 会将 p' 和 $\phi(p')$ 中的令牌移至 终点$_N$ 和 终点$_M$。

接下来，假设 $m \in \mathcal{R}(S, i_S)$ 是一个可达标识，且存在某个$p \in P_M$, $m(p) = m(\phi(p)) = 1$，并且 $m(q) = 0$ 对所有 $q \notin \{p, \phi(p)\}$ 均成立。假设 $u_1 \dots u_{k+1}$，则在 M 的骨架中，将令牌从库所 p 移至 终点$_M$ 的最短发射序列长度为 $k + 1$。

我们根据 u_1 的方向区分两种情况：

- $\lambda(u_1) = send$。首先注意到，u_1 在 m 中也处于使能状态。在 S 中发射 u_1，会在某个唯一的库所 $p' \in P_M \cap u_1{}^\bullet$ 中产生一个令牌。由于 $\lambda(u_1) = send$，发射 u_1 后，唯一的接口库所 $i \in u_1{}^\bullet \setminus P_M$ 中也会有一个令牌。因此，我们到达标识 m'，使得对于所有 $q \notin \{p', \phi(p)\ i\}$，都有 $m'(p') = m'(\phi(p)) = m'(i) = 1$，且 $m'(q) = 0$。

  由于 $\phi(p)$ 是 p 的镜像，我们发现 $\exists \bar{u}_1 \in T_N :\ \bar{u}_1 = \phi(u_1)$ 已启用，因为 $\{\phi(p)\ i\} = {}^\bullet\bar{u}_1$。发射 $\bar{u}_1$ 会消耗 $\phi(p)$ 中的令牌，并

i，在 $\phi(p')$ 中生成一个令牌。因此，我们到达一个标识 m''，使得对于所有 $q \notin \{p', \phi(p')\}$，$m''(p') = m''(\phi(p')) = 1$，且 $m''(q) = 0$。由于从 p' 到 fin_N 的最短发射序列 $u_2 \dots u_{k+1}$ 长度为 k，根据归纳法，我们发现，在 S 中发射 $(u_2 \langle\rangle \phi(u_2)) \dots (u_{k+1} \langle\rangle \phi(u_{k+1}))$ 会将 p' 中的令牌以及 $\phi(p')$ 移至 fin_M 和 fin_N。然而，接着发射 $(u_1 \langle\rangle \phi(u_1)) \dots (u_{k+1} \langle\rangle \phi(u_{k+1}))$ 会将 p 中的令牌以及 $\phi(p)$ 移至 fin_M 和 fin_N，即终止标识。

- 情形 $\lambda(u_1) = receive$ 的论证完全对偶。

因此，我们得出结论：引理由我们证明的命题直接推出。□

现在我们可以正式证明定理 1：

证明（定理 *1*）。设 N、M 和 S 如前所述。设 $m \in \mathcal{R}(S, i_S, f_S)$。

根据命题 1，必定存在一个变迁序列 $t_1, \dots\dots, t_n$，使得$m \xrightarrow{t_1} \dots \xrightarrow{t_n} m'$ 使得 $\lambda(t_i) = receive$ 对所有 $i \in 1 \dots n$ 都成立，并且存在一个库所$p \in P_M$，使得 $m'(p) = m'(\phi(p)) = 1$，且 S 中的所有其他库所均为空。由命题 2 可知，f_S 可达。因此，S 弱终止。□

通过证明以下性质，前述结果得到了加强：如果良构带标号端口网与其良构部分镜像的组合的终止库所有标识，那么该组合的其他库所均无标识。

定理 2（正确完成）。设 N 是一个良构带标号端口网，且 (M, ϕ) 是 N 的一个部分镜像。设 $S =$复合$(\{N, M\})$，且 $i_S = i_N + i_M$ 是 S 的初始标识，$f_S = f_N + f_M$ 是 S 的终止标识。对于网系统 (S, i_S, f_S) 中所有可达标识，若 N 和 M 的终止库所均有标识，则所有其他库所均为空。

证明。假设 N 和 M 的终止库所均有标识，而网 S 的其他库所中还留有令牌。N 和 M 的内部库所中不可能有剩余令牌，因为它们是状态机网，且各自的初始标识中都只有一个令牌（位于初始库所）。若接口库所中有剩余令牌，则根据定理 1，必定存在一条发射序列能够到达终止标识 f_S。这是不可能的，因为 N 和 M 的终止库所后集为空，矛盾。□

6 多个 Portnet 组合的弱终止

实际上，多个客户端可能希望反复使用服务器提供的同一服务。我们在上一章节中得到的结果无法原样推广到这种情形。为说明这一点，请看图 7 中所示的良构标记端口网 X, Y 和 Z。注意，(Y, ϕ_1) 和 (Z, ϕ_2) 是 X 的部分镜像。虽然组合 $compose(\{X, Y\})$ 和 $compose(\{X, Z\})$ 不会发生死锁，但组合 $compose(\{X, Y, Z\})$ 中的发射序列 $\langle 8, 10, 16, 2, 4, 6, 18, 19\rangle$ 会导致死锁。主要原因是客户端可能彼此干扰。事实证明，要实现预期结果，我们必须将这些结果置于一种同步模式中。

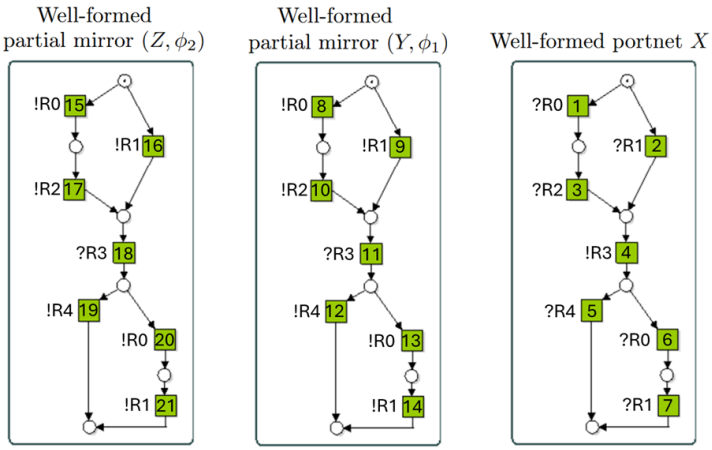


图 7：组合中的死锁：复合($\{X, Y, Z\}$)

同步模式要求服务器首先从希望与之协作的客户端中进行选择。此类协商可以通过图 8 所示的模式来建模。从初始标识开始，一个或多个客户端可以发送请求，服务器接收其中一个请求，随后发回响应。该响应恰好由一个客户端接收，这保证恰好一个客户端能够与服务器继续协作，最终在不受其他客户端干扰的情况下，为各自的终止库所标上标识。一旦服务器的终止库所被标上标识，闭合变迁 t 就会将令牌送回初始库所，此后服务器便可以选择下一个客户端。此过程会不断重复，直到所有客户端都得到处理并到达终止标识。

定义 15（同步模式）。图 8（左）所示的模式是一个同步模式$(\mathcal{C}, S, Q)$，其中 S 是良构带标号端口网 N 的闭包，$\mathcal{C} = \{C_1, C_2, \ldots C_n\}$ 是 N 的一组 n（$n \in \mathbb{N}$）良构镜像客户端，而 Q 是 $\{q, q_1, q_2, \ldots q_n\}$ 的可细化库所集合。网系统定义为(复合$(\mathcal{C} \cup \{S\})$ m_0, m_f)，其中 m_0 是由 $\{i\} \cup \{i_j \mid j \leq n\}$ 给出的初始标识，终止标识 m_f 由 $\{i\} \cup \{r_j \mid j \leq n\}$ 给出。

引理 6。该同步模式是弱终止的。

证明。根据定理 3.3，10 [7], 多个客户端端口网与一个服务器端口网的组合是弱终止的。同步模式正是这样一种组合，因此它也弱终止。□

同步模式的结构带来以下观察结果：首先，由于客户端和服务器的骨架都是在初始化中只有一个令牌的 S 网，因此其内部库所中至多有一个令牌。其次，服务器可以消耗某个客户端产生的恰好一个令牌。随后，服务器可以向其客户端返回恰好一个令牌。第三，至多有一个

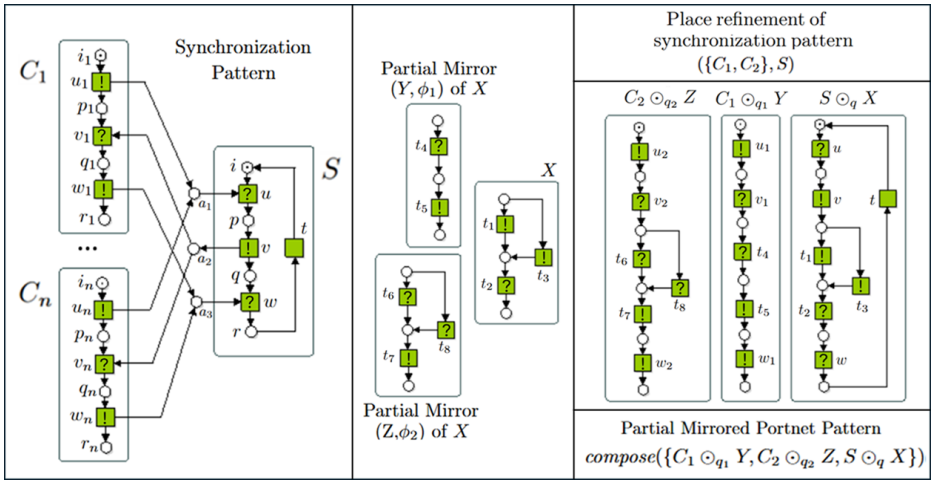


图 8：同步模式和部分镜像端口网模式

客户端消耗服务器产生的一个令牌，并成为被选中的客户端。选中某个客户端后，该客户端和服务器的镜像库所会被标识（先标识服务器，再标识客户端），并且一方产生的所有接口令牌都被另一方消耗。从此处开始，被选中的客户端产生一个令牌并标识其终止库所，从而使服务器能够取用该令牌并标识其终止库所。

引理 7。 设 m 是一个同步模式的标识，具有 $n \in \mathbb{N}$ 个客户端，并满足以下性质

1. $m(i) + m(p) + m(q) + m(r) = 1$2. $\forall j \in \{1 \dots n\}: m(i_j) + m(p_j) + m(q_j) + m(r_j) = 1$3. $\sum_{j \in \{1 \dots n\}} m(p_j) = m(a_1) + m(p) + m(a_2)$,
4. $(\sum_{k \in \{1 \dots n\}} m(q_k)) + m(a_2) \leq 1$, 5. $m(a_2) + m(a_3) \leq 1$ 和 $m(a_2) + m(a_3) = m(q) - \sum_{k \in \{1 \dots n\}} m(q_k)$

那么，发射任意使能变迁 t *from* m 都不会违反这些性质。

证明。 假设 m 是满足所述性质的一个标识。

– 假设变迁 u 已使能。那么 $m(i) > 0 \wedge m(a_1) > 0$。变迁 u的发射会从 i 和 a_1 各消耗一个令牌，并向 p添加一个令牌。这会使不变量 1 和 3 中各库所的令牌数发生如下变化

$$m(i) - 1 + m(p) + 1 + m(q) + m(r) = 1$$

and

$$\sum_{j \in \{1, \dots, n\}} m(p_j) = m(a_1) - 1 + m(p) + 1 + m(a_2)$$

因此，发射 u 会产生一个再次满足这些不变量的标识。

– 假设变迁 v 是使能的。那么 $m(p) > 0$。根据不变量 1，我们推导出 $m(q) = 0$。因此，在不变量 5 中，$m(q) - \sum_{k\in\{1,\ldots,n\}} m(q_k) = 0$，这意味着 $m(a_2) + m(a_3) = 0$。

变迁 v的发射会从 p 消耗一个令牌，并向 q和 a_2各添加一个令牌。这会使不变量 1、3、4 和 5 中库所的令牌数发生如下变化

$$m(i) + m(p) - 1 + m(q) + 1 + m(r) = 1$$

$$\sum_{j\in\{1,\ldots,n\}} m(p_j) = m(a_1) + m(p) - 1 + m(a_2) + 1$$

$$\sum_{k\in\{1,\ldots,n\}} m(q_k) + m(a_2) + 1 \leq 1$$

$$m(a_2) + 1 + m(a_3) \leq 1$$

$$m(a_2) + 1 + m(a_3) = m(q) + 1 - \sum_{k\in\{1\ldots n\}} m(q_k)$$

因此，发射 u 会得到一个标识，使这些不变量再次成立。

– 假设变迁 w 使能。那么 $m(q) > 0$，且 $m(a_3) > 0$。根据不变量 5，我们可以得出 $m(a_2) = 0$。变迁 w的发射会从 q 和 a_3 各消耗一个令牌，并向 r添加一个令牌。这会使不变量 1 和 5 中库所的令牌数发生如下变化

$$m(i) + m(p) + m(q) - 1 + m(r) + 1 = 1$$

$$m(a_2) + m(a_3) - 1 \leq 1$$

$$m(a_2) + m(a_3) - 1 = m(q) - 1 - \sum_{k\in\{1,\ldots,n\}} m(q_k)$$

因此，发射 w 会得到一个标识，使这些不变量再次成立。

– 假设变迁 t 使能。那么 $m(r) > 0$。变迁 t的发射会从 r 消耗一个令牌，并向 i添加一个令牌。这会使不变量 1 中库所的令牌数发生如下变化

$$m(i) + 1 + m(p) + m(q) + m(r) - 1 = 1$$

因此，发射 t 会得到一个再次满足这些不变量的标识。

– 假设对于某个 $j \in \{1, \ldots, n\}$，变迁 u_j 已使能。则 $m(i_j) > 0$。发射 u_j 会从 i_j 消耗一个令牌，并向 a_1 和 p_j 各添加一个令牌。这会使不变量 2 和 3 中库所的令牌数量发生如下变化

$$m(i_j) - 1 + m(p_j) + 1 + m(q_j) + m(r_j) = 1$$

$$(\sum_{j\in\{1,\dots,n\}} m(p_j)) + 1 = m(a_1) + 1 + m(p) + m(a_2)$$

因此，发射 u_j 会得到一个再次满足这些不变量的标识。

– 假设对于某个 $j \in \{1 \dots n\}$，变迁 v_j 处于使能状态。那么 $m(a_1) > 0 \wedge m(p_j) > 0$。变迁 v_j 的发射会从 p_j 和 a_2 中各消耗一个令牌，并向 q_j 中添加一个令牌。这会使不变量 2、3、4 和 5 中库所的数量发生如下变化

$$m(i_j) + m(p_j) - 1 + m(q_j) + 1 + m(r_j) = 1$$

$$(\sum_{j\in\{1,\dots,n\}} m(p_j)) - 1 = m(a_1) + m(p) + m(a_2) - 1$$

$$(\sum_{j\in\{1,\dots,n\}} m(q_j)) + 1 + m(a_2) - 1 \leq 1$$

$$m(a_2) - 1 + m(a_3) = m(q) - (\sum_{j\in\{1,\dots,n\}} m(q_j) + 1)$$

因此，发射 v_j 会得到一个标识，使这些不变量再次得到满足。

– 假设对于某个 $j \in \{1, \dots, n\}$，变迁 w_j 处于使能状态。那么 $m(q_j) > 0$。根据不变量 4，我们可以推知 $m(a_2) = 0$。此外，根据不变量 1 和 5，我们可以推知 $m(q) = 1$。变迁 w_j 的发射会从 q_j 中消耗一个令牌，并向 r_j 和 a_3 中各添加一个令牌。这会使不变量 2、4 和 5 中库所的令牌数量发生如下变化

$$m(i_j) + m(p_j) + m(q_j) - 1 + m(r_j) + 1 = 1$$

$$(\sum_{k\in\{1,\dots,n\}} m(q_k) - 1) + m(a_2) \leq 1$$

$$m(a_2) + m(a_3) + 1 \leq 1$$

$$m(a_2) + m(a_3) + 1 = m(q) - (\sum_{k\in\{1,\dots,n\}} m(q_k) - 1)$$

因此，发射 w_j 会产生一个再次满足这些不变量的标识。

由此我们得出结论：从满足全部 5 项性质的标识发射任意使能变迁，所得到的标识仍满足这些性质。□

推论 2. 具有 $n \in \mathbb{N}$ 客户端的同步模式的网系统满足

- $\forall m \in \mathcal{R}(N, m_0) : m(q) = 1 \implies \sum_{j\in\{1\dots n\}} m(q_j) \leq 1$
- $\forall m \in \mathcal{R}(N, m_0) : \sum_{j\in\{1\dots n\}} m(q_j) = 1 \implies m(q) = 1$

证明。 由引理 7 以及标识 m_0 满足库所不变量（定义 15）这一事实可得。 □

库所和变迁的细化 [16] 是一种广为人知的策略，可以在保持佩特里网健全性的同时，逐步细化其行为。我们将经典的库所细化操作扩展到带标签 OPN。

定义 *16*（带标签 *OPN* 的库所细化）。给定两个互不相交的带标签 *OPN* N和 M。用 M细化库所 $p \in P_N$，记作$N \odot_p M$，所得带标签 *OPN* 为 $(P, I, O, T, F,$ 初始化$, \mathit{fin}\ \mu)$，其中 $P = (P_N \setminus \{p\}) \cup P_M$,$I = I_N \cup I_M$, $O = O_N \cup O_M$, $T = T_N \cup T_M$, $F =(F_N \setminus (({}^\bullet p \times \{p\}) \cup (\{p\} \times p^\bullet))) \cup F_M \cup ({}^\bullet p \times$ 初始化$_M) \cup (\mathit{fin}_M \times p^\bullet)$, 初始化 $=$初始化$_N$, fin $=\mathit{fin}_N$, $\mu = \mu_N \cup \mu_M$。

我们形式化地引入部分镜像端口网模式的概念。这种模式通过库所细化嵌入同步模式中，捕捉了服务器与多个客户端通信的本质，并由对服务器进行（部分）镜像而获得。图 8 展示了一个示例。

定义 *17*（部分镜像端口网模式）。设 N 是一个弱终止的良构标记端口网，并设 $(M_1, \phi_1) \ldots, (M_n, \phi_n)$（对于 $n \in \mathbb{N}$）是 N 的一组互不相交的部分镜像。设 $R = (\mathcal{C}, S, \{q, q_1, \ldots q_n\})$ 是一个具有 n 个客户端的同步模式，使得 $M_1, \ldots, M_n, N, \mathcal{C}, S$ 具有互不相交的骨架。部分镜像端口网模式是一个复合 $L = \mathit{compose}(\mathcal{C}' \cup \{\mathcal{S}'\})$，其中 $\mathcal{C}' = \{C_1 \odot_{q_1} M_1, \ldots, C_n \odot_{q_n} M_n\}$ 和 $\mathcal{S}' = S \odot_q N$ 具有初始标识 $m_0 = i_S + i_{C_1} + \ldots + i_{C_n}$、终止标识 $m_f = i_S + f_{C_1} + \ldots + f_{C_n}$ 以及网系统 (L, m_0, m_f)。如果 $M_1, \ldots, M_n$ 是 N 的完整镜像，则称 L 为镜像端口网模式。

定理 3. 部分镜像端口网模式弱终止。

证明。 由引理 6 可知，由 $n \in \mathbb{N}$个客户端组成的同步模式弱终止。由推论 2 可知，服务器需要先为库所 q添加标识，然后对于某个 $j \in \{1 \ldots n\}$，恰好一个客户端 C_j才能为其库所 q_j添加标识。此外，在 q 被标记之前，服务器不能从其中移除令牌。在服务器的骨架中，库所 q 始终位于从初始库所 i 到终止库所 r 的一条路径上。因此，同步端口网中从初始标识到终止标识的所有发射序列，都会先为库所 q 添加一个令牌，再为库所 q_j添加令牌，同时 q 保持有标识。

根据定义 16 和定义 17，良构带标号端口网 N及其部分镜像（M_1，ϕ_1...，...，（M_n，ϕ_n））中，只有初始库所和终止库所的前集和后集中包含同步模式的节点（变迁）。因此，同步模式 R 的一个标识若在 q和 q_j中各包含一个令牌，则对应于部分镜像端口网模式 L 中的一个标识，其中 N 和 M_j的初始库所中分别有一个令牌。根据定理 1，我们知道，从这种标识可达的所有标识都存在一条通向终止标识的发射序列，即通向 N 和 M_j的终止库所。此外，根据定理 2，我们知道，当 N 和 M_j 的终止库所中有标识时， N 和 M_j 中的所有其他库所均为空。根据定义 16 和定义 17， L 中包含

一个令牌位于 N 和 M_j 的终止库所中（且 S的库所中没有其他令牌）时，所启用的变迁集合与网 R 在库所 q 和 q_j中各有一个令牌时所启用的变迁集合完全相同。因此，使用多客户端端口网进行细化不会破坏同步模式的弱终止性质。□

与组件框架的关系。 [5] 中介绍的基于组件的体系结构框架建立在两个概念之上：组件和交互模式。组件 是一个具有名为“空闲”的特殊库所的强连通开放 Petri 网，这意味着组件在执行过程中是循环的，因此能够处理多个客户端。交互模式 是一类参数化的多工作流网，由一组相同的客户端（OWN）和一组相同的服务器（组件）组合而成。文中定义了四种反复出现且弱终止的交互模式，分别是远程过程调用（RPC）、发布-订阅（PS）、目标-反馈-结果（GFR）和镜像端口（MP）。为了将系统设计表示为基于交互模式的组件组合，文中提出了一种基于图的体系结构图表示法。为了从体系结构图推导出佩特里网，文中规定了一种构造方法，并证明该方法能够保证弱终止性质。该构造方法是一种细化策略，基于同时细化 和插入 操作，将交互模式逐步引入系统（开放 Petri 网）中。

要在此框架中添加一种新模式，只需证明它是弱终止的交互模式，因为 [5, 引理 3.4.1] 保证同时细化会保持弱终止性。由定理 3 可知，镜像端口网模式是弱终止的。同步模式是 MP 模式的一个实例，而 MP 模式是一种交互模式。镜像端口网模式是通过使用 S-OWN 对同步模式中可细化库所进行库所细化得到的，这些 S-OWN 对应于一个良构服务器及其镜像客户端；随后将它们组合，所得结果仍然是一种交互模式。

命题 3（同步模式的细化）。镜像端口网模式是一种交互模式。

证明。根据定义 15，同步模式是一种不含抑制弧和重置弧的交互模式。根据定义 7，良构服务器端口网与其完整镜像客户端的组合满足：（i）对于任意客户端与服务器对，客户端的输入库所等于服务器的输出库所，反之亦然；（ii）连接到同一接口库所的客户端与服务器变迁方向相反。根据定义 17，使用良构服务器端口网及其完整镜像客户端，对同步模式中的可细化库所进行连续的库所细化（定义 16）后，得到的仍然是一个服务器（它是一个 S-OWN 的闭包）和一组 S-OWN 客户端。它们的组合是一种不含抑制弧和重置弧的交互模式。□

7 应用

人们日益认识到，缺乏明确、精确的组件接口规范可能会导致系统集成期间出现问题，这

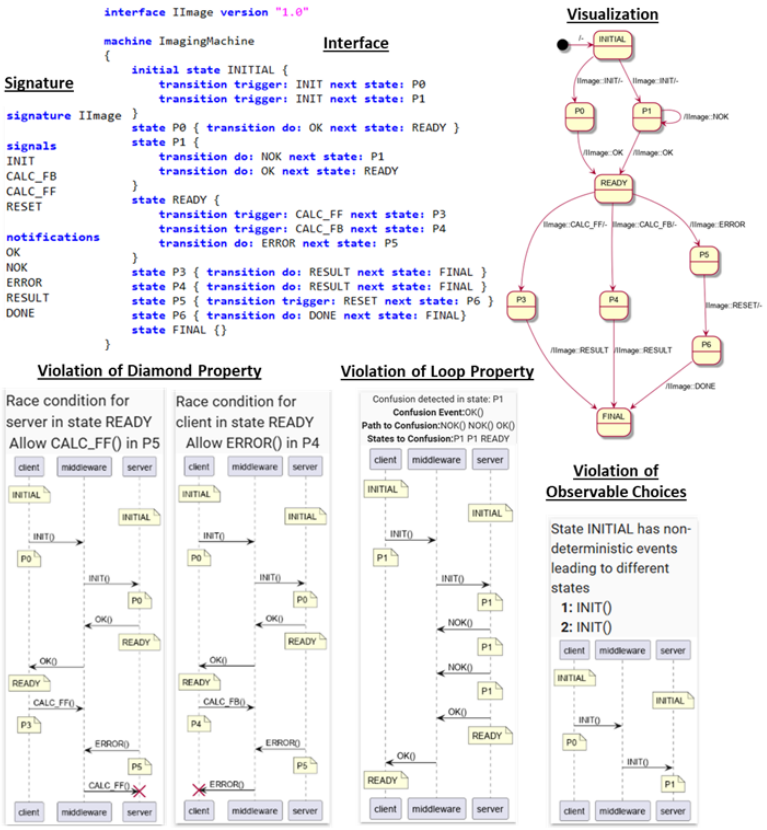


图9：将结果应用于 ComMA

促使人们开发了开源接口建模工具 ComMA [13]。其目标是捕获接口的静态和动态方面，为服务提供者与消费者之间的形式化契约奠定可靠基础。ComMA 规范用于借助运行时监控 [12] 以及近来采用的基于模型的测试技术，验证实现是否符合其规范。该工具是在与业界合作的过程中开发的，Philips 和 Thales 的软件工程师将其用于各自的开发流程。为充分发挥其优势并尽早发现设计错误，人们认识到接口规范必须正确。本文的研究成果已作为算法集成到 ComMA 中，用于检查接口规范的良构性。事实证明，这项功能有助于提高人们对设计异步系统时可能遇到的陷阱的认识，并可指导工程师从设计之初就构建弱终止接口协议。ComMA 提供一系列领域特定语言，包括类型、签名、接口和输入参数。类型 用于描述由基本类型（int、real、string 等）、枚举、向量、记录和映射构成的数据定义。签名-

性质 用于描述以下类型的事件定义（即其标签和数据参数类型）：命令-回复（客户端发送命令，并在服务器回复前阻塞）、信号（客户端发送非阻塞事件）和通知（服务器发送非阻塞事件）。接口 描述输入和输出事件允许的顺序，这些事件作为协议状态机的变迁定义，从服务提供者（服务器）的视角出发，也称为提供接口。状态机还可以定义全局变量，并使用表达式语言对其进行操作。包含数据参数的输出事件引用全局变量，而输入事件的数据参数则由客户端提供。在定义另一组件如何使用提供接口时，通常只需镜像服务器的协议。因此，会省略客户端协议的显式模型。为了对服务器状态机进行独立模拟，输入参数用于指定输入事件参数在各个可能状态下可能取值。通过将服务器状态机转换为有色Petri网，便可对其进行分析和模拟，该转换使用 SNAKES [17] 实现。本文不讨论该转换的形式化处理。本节其余部分将略去数据方面的内容，并非正式地讨论到带标记端口网的映射。

图9展示了图3中服务器端口网 S 的 ComMA 规范。签名定义了信号和通知，其名称分别对应于 S的变迁标签。接口状态机的语法由状态标签（包括显式初始状态）组成，变迁则是状态之间的有向边。图9展示了以 UML 语法生成的状态机可视化图。每个状态都映射到底层端口网中的一个库所，其名称与该状态的标签相对应。每个状态定义了一组可能的变迁，每个变迁定义一个目标<e
m style='3'>下一状态。变迁分为两类。非触发迁移定义了服务器在do 章节中的主动权，该章节包含一个通知，该通知映射为一个发送变迁，其前集对应于该变迁所定义的状态（库所），后集对应于下一状态所指向的状态（库所）。触发迁移定义了客户端在$trigger$ 章节中的主动权，该章节包含一个信号，该信号映射为一个接收变迁，其前集对应于该变迁所定义的状态（库所），后集对应于下一状态所指向的状态（库所）。根据这些映射规则，可以从状态机语法推导出带标号端口网。底层转换会丢弃接口库所，因为我们只关注骨架的分析。对于紧凑规范，还提供了许多简写符号，例如，包含多个do 章节的触发迁移，每个章节包含一个通知序列；$for\ all\ states$ 可避免在多个状态中重复相同的变迁，等等。所有构造都可以用迄今为止介绍的概念来表达。

为帮助建模人员创建弱终止规约，我们在 ComMA 中实现了相关算法：首先验证骨架是否弱终止，然后验证可观测选择、菱形性质和循环性质是否成立。检测到违反条件的情况时，系统会生成一张统一建模语言序列图（见图9），展示服务器与一个假定完全镜像该协议的客户端组合时出现的问题。

8 结论

本文提出了保证服务器与多个部分镜像客户端协议组合弱终止的充分条件。这些条件通过良构性施加结构限制，放宽了现有研究所施加的严格限制，因此涵盖了实践中遇到的更大范围的通信协议。在 ComMA 工具中嵌入服务器协议的良构性检查，并假设客户端采用对称镜像后，建模人员便能在设计早期了解组件交互中可能存在的潜在隐患。作为未来工作，我们计划推广为保持多个客户端组合弱终止而引入的同步模式。我们还计划将研究成果扩展到 ComMA 的组件建模语言，其中函数约束用于限定所提供和所需接口上的事件。

致谢

本研究是*TechFlex 2026* 和*Matala 2025* 计划的一部分，由 TNO-ESI 负责，并与 ASML Holding N.V. 和 ThalesnederlandB.V. 合作开展。研究活动由 Holland High Tech | TKI HTSM 通过公私合作伙伴关系的 PPP创新计划（PPP-I）共同资助。

参考文献

1. van der Aalst, W.M., van Hee, K.M., Massuthe, P., Sidorova, N., van der Werf, J.M.: 组合式服务树。载于：Petri 网的应用与理论，第30届国际会议，PETRI NETS 2009，法国巴黎，2009年6月22日至26日。会议论文集。计算机科学讲义，第5606卷，第283–302页。施普林格（ 2009 10 1007 978 3 642 02424 5_17 ）。 https://doi.org/ 。/ - - - - 。Amat, N., Amparore, E., Berthomieu, B., Bouvier, P., Zilio, S.D., Hulin-Hubard, F., Jensen, P.G., Jezequel, L., Kordon, F., Li, S., Paviot-Adet, E., Petrucci, L., Srba, J., Thierry-Mieg, Y., Wolf, K.：模型检测竞赛幕后：对2018年至2023年结果的分析。载于：Beyer, D., Hartmanns, A., Kordon, F.（编）TOOLympics挑战赛 2023，第52–89页。施普林格自然瑞士（2025）。3. Baldan, P., Bonchi, F., Gadducci, F., Monreale, G.V.：使用开放 Petri 网对同步和异步交互进行模块化编码。《计算机编程科学》**109**，96–124（2015）。https://doi.org/10.1016/J.SCICO.2014.11. 019 4. Baldan, P., Corradini, A., Ehrig, H., Heckel, R.：基于确定性过程的开放 Petri 网组合语义。《计算机科学中的数学结构》**1**5096012950400431 1 5 1 1 35 2005 10 1017 ( ), – ( )。 https://doi.org/ . /S . Bera, D.：用于机器人建模的 Petri 网。博士论文，埃因霍温理工大学（2014）。https://doi.org/10.6100/IR780942

6. Bera, D., van Hee, K.M., van der Werf, J.M.：设计弱终止 ROS 系统。载于：Petri 网的应用与理论——第33届国际会议，PETRI NETS 2012，汉堡，德国，2012年6月25日至29日。会议论文集。计算机科学讲义，第7347卷，第328–347页。施普林格（2012）。https://doi. org/ 10 1007 978 3 642 31131 4_18 。/ - - - - 。Bera, D., Van Hee, K.M., Van Osch, M., van der Werf, J.M.E. 等：一种端口兼容性蕴含弱终止的组件框架。载于：Duvigneau, M., Moldt, D., Hiraishi, K.（编）Petri 网与软件工程国际研讨会论文集，泰恩河畔纽卡斯尔，英国，2011年6月20日至21日，第723卷，第152–166页（2011）。CEUR 研讨会论文集。CEUR-WS.org，https://ceur-ws.org/Vol-723/paper11.pdf 8. De Alfaro, L., Henzinger, T.A.：接口自动机。载于：Tjoa, A.M., Gruhn, V.（编）第8届欧洲软件工程会议与第9届 ACM SIGSOFT 软件工程基础国际研讨会联合会议论文集，维也纳，奥地利，2001年9月10日至14日。第109–120页。ACM（2001 503226 9 10 1145 503209）。https://doi.org/ . / . . Desel, J., Esparza, J.：自由选择 Petri 网。剑桥理论计算机科学论丛，剑桥大学出版社（1995）。https://doi.org/10. 1017 9780511526558 10 /CBO . van Hee, K.M., Sidorova, N., Voorhoeve, M.：逐步细化方法中工作流网的健全性与可分离性。载于：van der Aalst, W.M.P., Best, E.（编）Petri 网的应用与理论 2003，第24届国际会议，ICATPN 2003，埃因霍温，荷兰，2003年6月23日至27日，会议论文集。计算机科学讲义，第2679卷，第337–356页。施普林格（2003）。https://doi.org/ 10 1007 3 540 44919 1_22 。/ - - - 。van Hee, K.M., Sidorova, N., van der Werf, J.M.：异步通信系统的构造：确保弱终止！载于：软件组合——第9届国际会议，SC@TOOLS 2010，马拉加，西班牙，2010年7月1日至2日。会议论文集。计算机科学讲义，第6144卷，第106–121页。施普林格（ 2010 10 1007 978 3 642 14046 4_8 ）。https://doi.org/ 。/ - - - - 。Kurtev, I., Hooman, J., Schuts, M.：基于接口规范的运行时监控。载于：ModelEd 、TestEd、TrustEd——纪念 Ed Brinksma 60岁生日的论文集，计算机科学讲义，第10500卷，第 335 356 2017 10 1007 978 3 319 68270 9_17 – 页。施普林格（）。https://doi.org/ 。/ - - - - 。Kurtev, I., Hooman, J., Schuts, M., van der Munnik, D.：使用 ComMA 进行基于模型的组件开发与分析。《计算机编程科学》**233**，103067（2024）。https://doi.org/ . /J.SCICO. .103067 14. McIlroy, M.D., Buxton, J., Naur, P., Randell, B.：量产软件组件。载于：第一届国际软件工程会议论文集，德国加米施-帕滕基兴。第88–98页（1968）。15. Milner, R.：面向 Petri 网的双图。载于：Desel, J., Reisig, W., Rozenberg, G.（编）《并发与 Petri 网讲座》，《Petri 网进展》 [本教程卷源自2003年9月在德国艾希施泰特举办的第4届 Petri 网高级课程（ACPN 2003）。]。计算机科学讲义，第3098卷，第686–701页。施普林格（2003）。https://doi.org/10.1007/ 978 3 540 27755 2_19 - - - - 。Murata, T.：Petri 网：性质、分析与应用。IEEE 会议录 **77**(4)，541 24143 17 580 1989 10 1109 5 – ( )。https://doi.org/ . / . . Pommereau, F.：SNAKES：灵活的高级 Petri 网库（工具论文）。载于：Devillers, R., Valmari, A.（编）Petri 网与并发的应用与理论——第36届国际会议，PETRI NETS 2015，比利时布鲁塞尔，

2015年6月21日至26日，会议论文集。计算机科学讲义，第9115卷，第
254 265 2015 10 1007 978 3 319 19488 2_13 –页。
18
施普林格（）。https://doi.org/。/----。Sassone, V., Sobocinski, P.：《佩特里网的一致性》。载于：Ehrig, H.、Padberg, J.、Rozenberg, G.（编），《佩特里网与图变换研讨会会议论文集》，PNGT@ICGT 2004，罗马，意大利，2004年10月2日。理论计算机科学电子笔记，第127卷，第107–120页。爱思唯尔（2004）。https://doi.org/10 008 19 1016 2005 02。/J.ENTCS.。。Szyperski, C.A.、Gruntz, D.、Murer, S.：《构件软件——超越面向对象编程》，第2版。Addison-Wesley 构件软件丛书，Addison-Wesley（2002 248041840 20），https://www.worldcat.org/oclc/。Wolf, K.：《我的服务有合作伙伴吗？》《佩特里网及其他并发模型汇刊》II：流程感知信息系统中的并发性特刊 **2**，152–171（2009）。https://doi.org/10.1007/978-3-642-00899-3_9